A Data-Center FPGA Acceleration Platform for Convolutional Neural Networks

Xiaoyu Yu[1], Yuwei Wang[1], Jie Miao[1], Ephrem Wu[2], Heng Zhang[1], Yu Meng[1], Bo Zhang[1], Biao Min[1], Dewei Chen[1], Jianlin Gao[1]

1 Tencent Shenzhen, China

2 Xilinx, Inc., San Jose, CA 95124, USA

Tencent 腾讯

XILINX®

About Tencent

Tencent is founded in

1998

One of the

Top 5

Internet Companies by Market Value

Monthly active users reach

1/0.8 Billion

for WeChat/QQ

Users in over

200

countries

Background

- CNN Models are widely used in Tencent
- Billions of operations per inference task × Billions of task each day
- Models are still in fast evolution.
- A reconfigurable accelerator is desirable
- Three key objectives:
  - Support different CNN models, easy to try
  - Achieve higher performance to lower TCO
  - Low latency

Framework for General Purpose CNN Acceleration

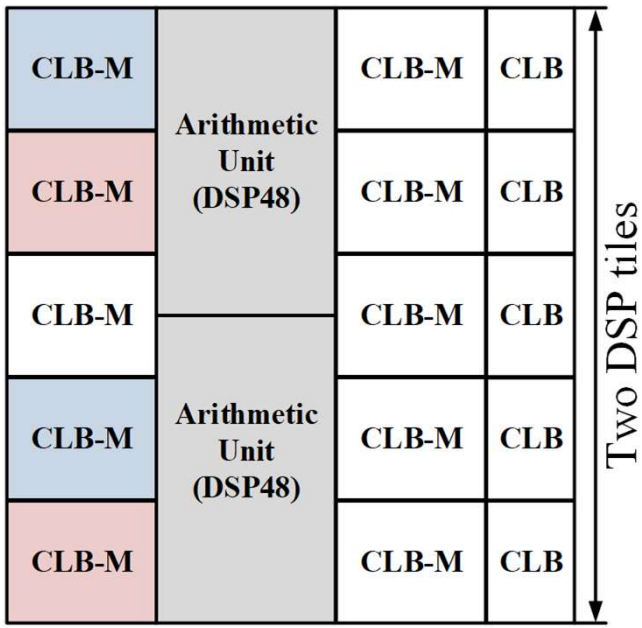
- More and more CNN models

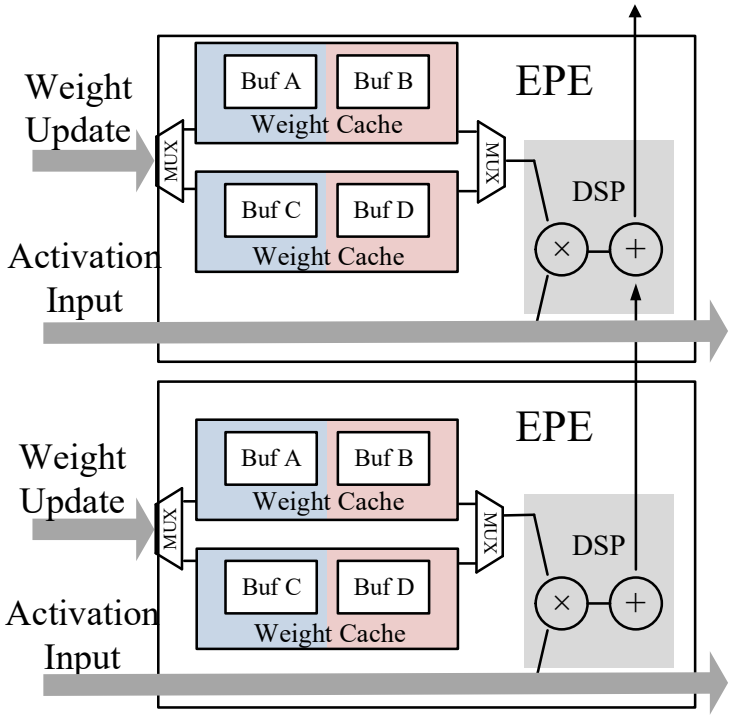
- Operator Classification
  - Convolution : 19% total types, 95%+ computation cost
  - Non-convolution : 81% total types, 5 %- computation cost

- Different design strategies
  - Convolution : Performance improvement
  - Non-convolution : Support mass types of operators

Unified Computing Engine for Convolution—Supertile

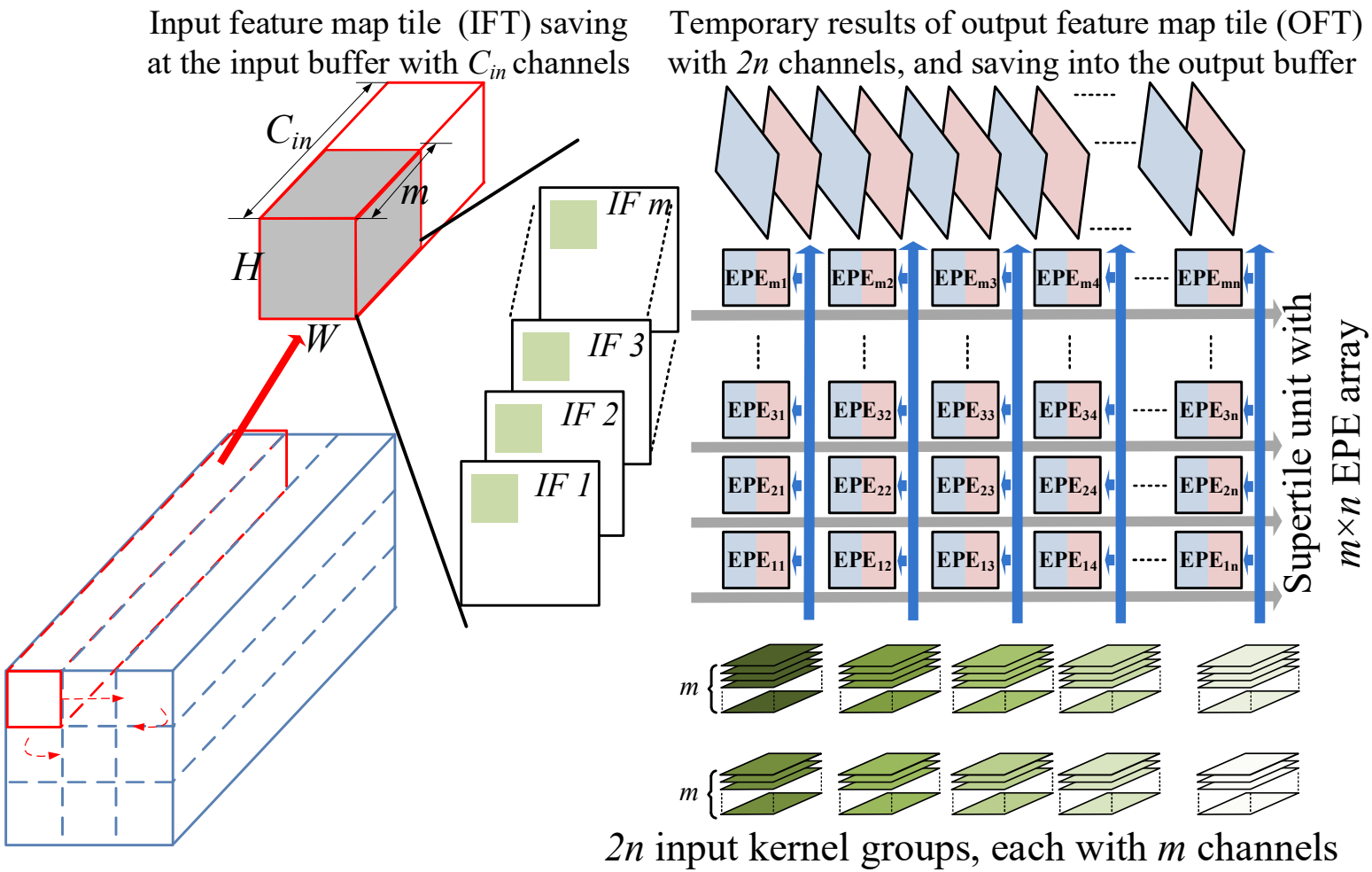
- Performance = Freq. * Dsp_num * Ops_per_dsp
- The supertile method runs the DSP at twice the clock rate of the surrounding logic[1].
- Enhanced Processing Element (EPE)





[1] E. Wu, X. Zhang, D. Berman, and I. Cho. “A high-throughput reconfigurable processing array for neural networks,” In Field Programmable Logic and Applications (FPL), 2017 27th International Conference on (pp. 1–4). IEEE.

Unified Computing Engine for Convolution -- Supertile Unit (SU)

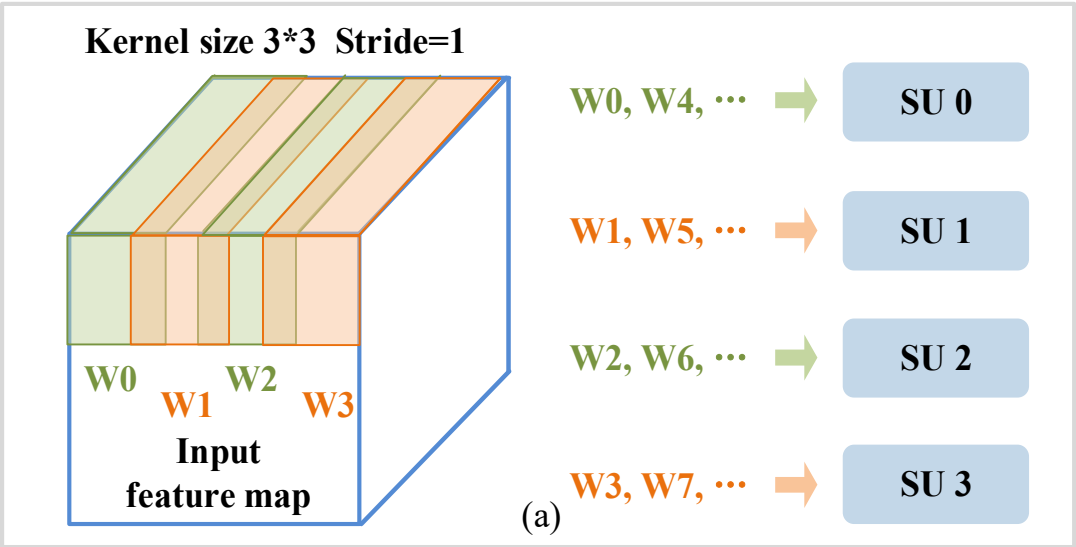


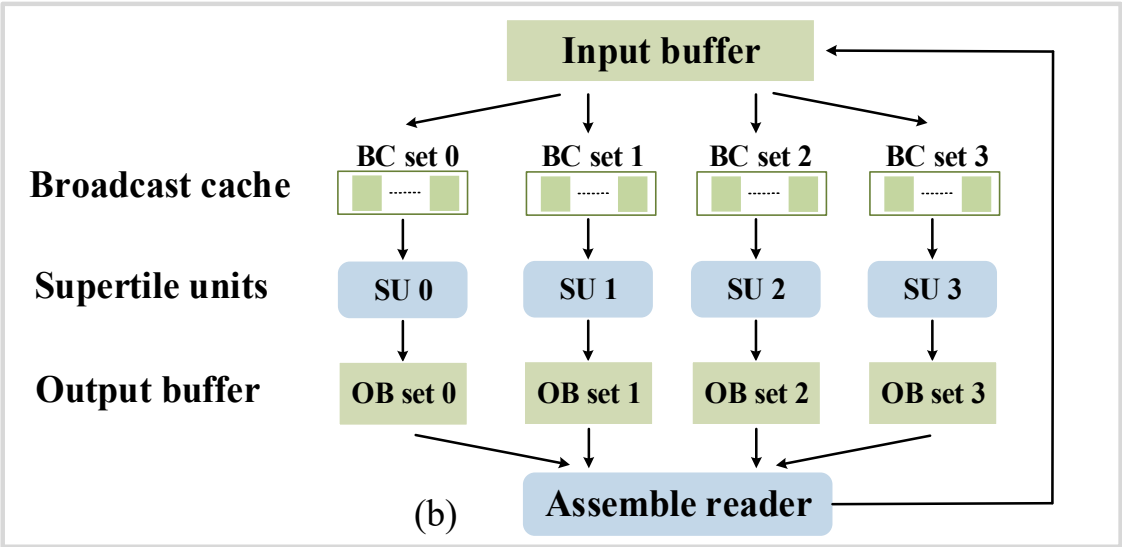
Convolution with an input feature map tile (IFT) and 2n kernel groups on one SU.

Unified Computing Engine for Convolution -- Scaled-up SU

- Two Challenges:
  - task partition.
  - data bandwidth would be multiplied

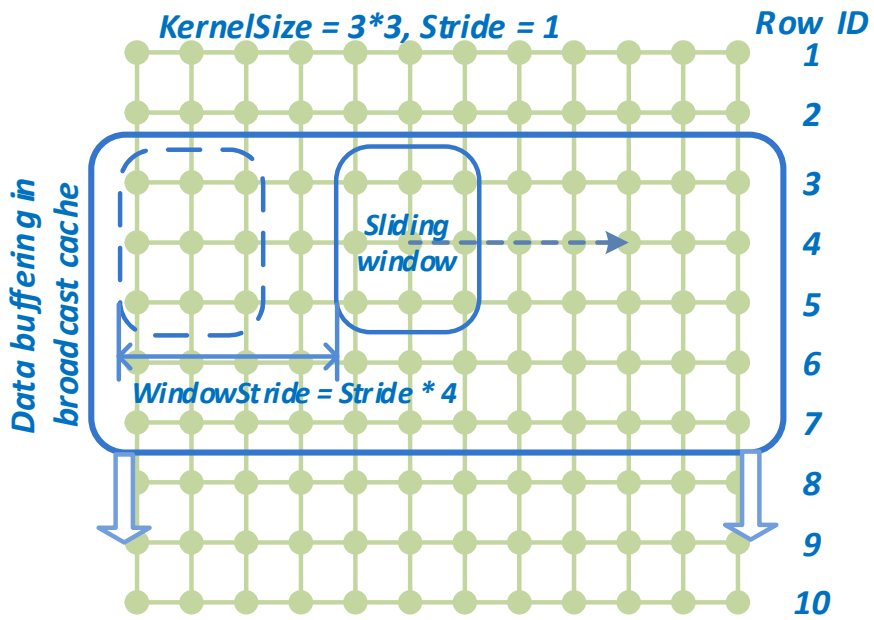
- Solutions:
  - Interleaved task dispatching
  - Dispatching-assembling buffering model
  - Broadcast cache (BC)

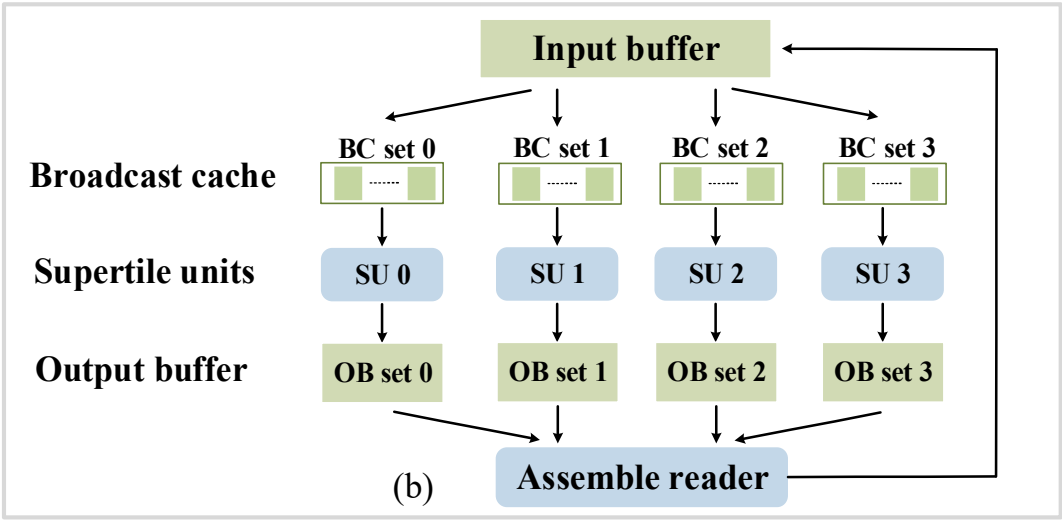




Unified Computing Engine for Convolution – Broadcast Cache

- a circular buffer
- BC-window-stride = 4 × *Convolution-Stride*
- Increase bandwidth from $512 \times f_{logic}$ bit/s into $2048 \times f_{logic}$ bit/s





Non-convolution Ops in inference

- Challenges:
  - Mass types of non-convolution ops
  - Resource limitation
- Solutions:
  - Perform different design strategy for different class
    - Filter Processing Unit (FPU)

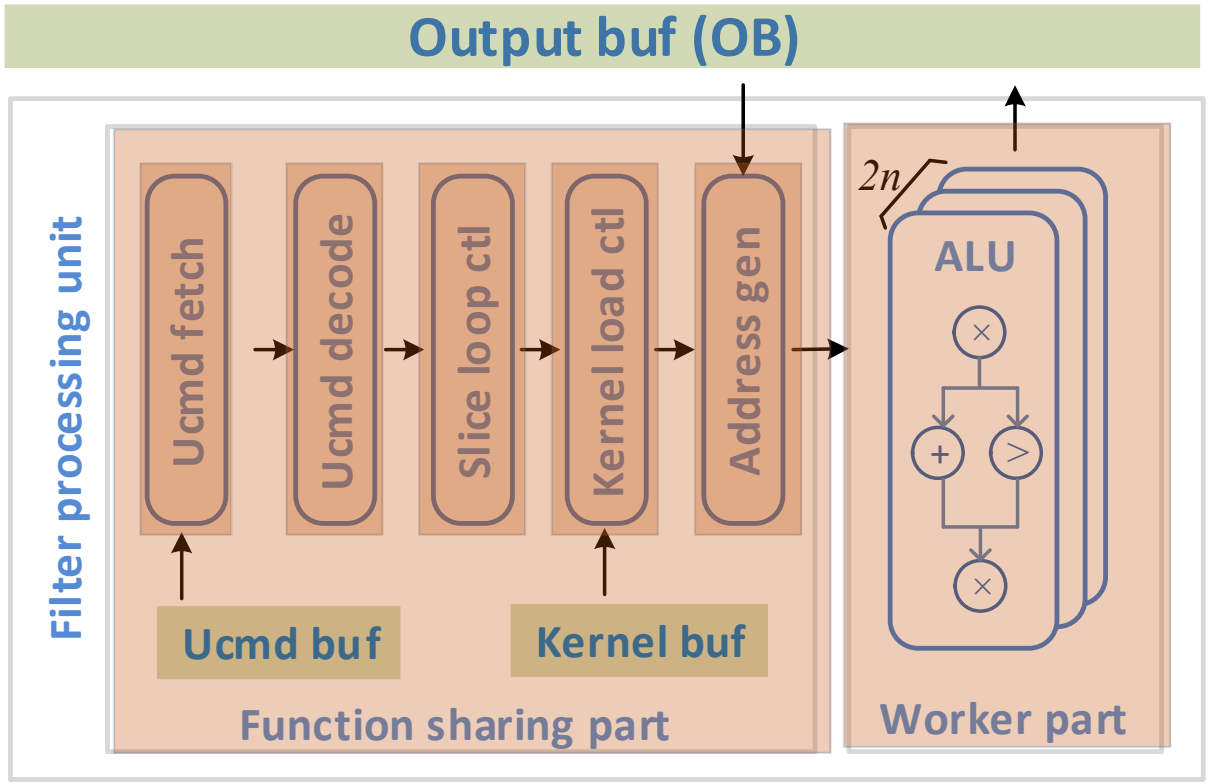
      MaxPool/AvgPool/DepthwiseConv/BN/Relu/Relu6
    - Customization: operations across channels

      LRN
    - Operator Fusion.

      ElementAdd/Relu/DynamicQuantization
  - Functional-logic-sharing

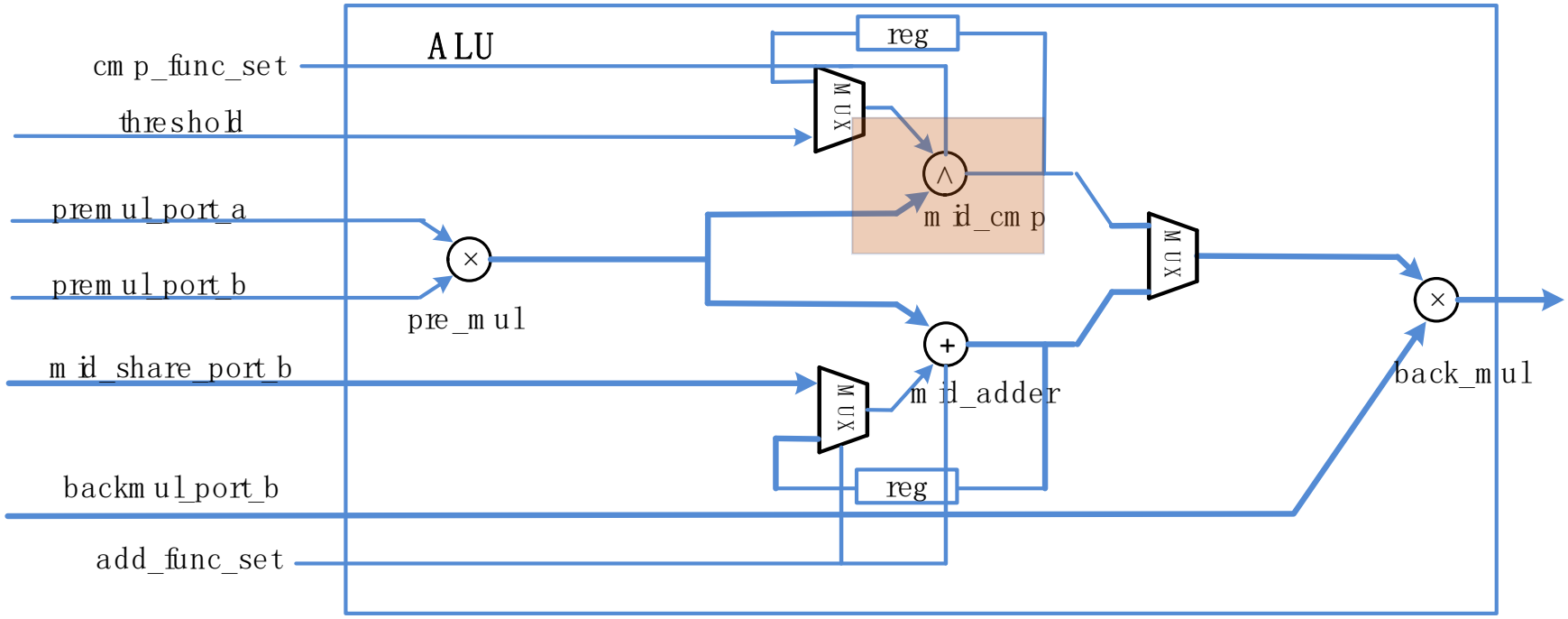
Postprocessing – FPU

- Common Styles :
  - Two level of data access style
  - no operations exist across channels
  - parameters similarities
  - Pointwise operations as special cases



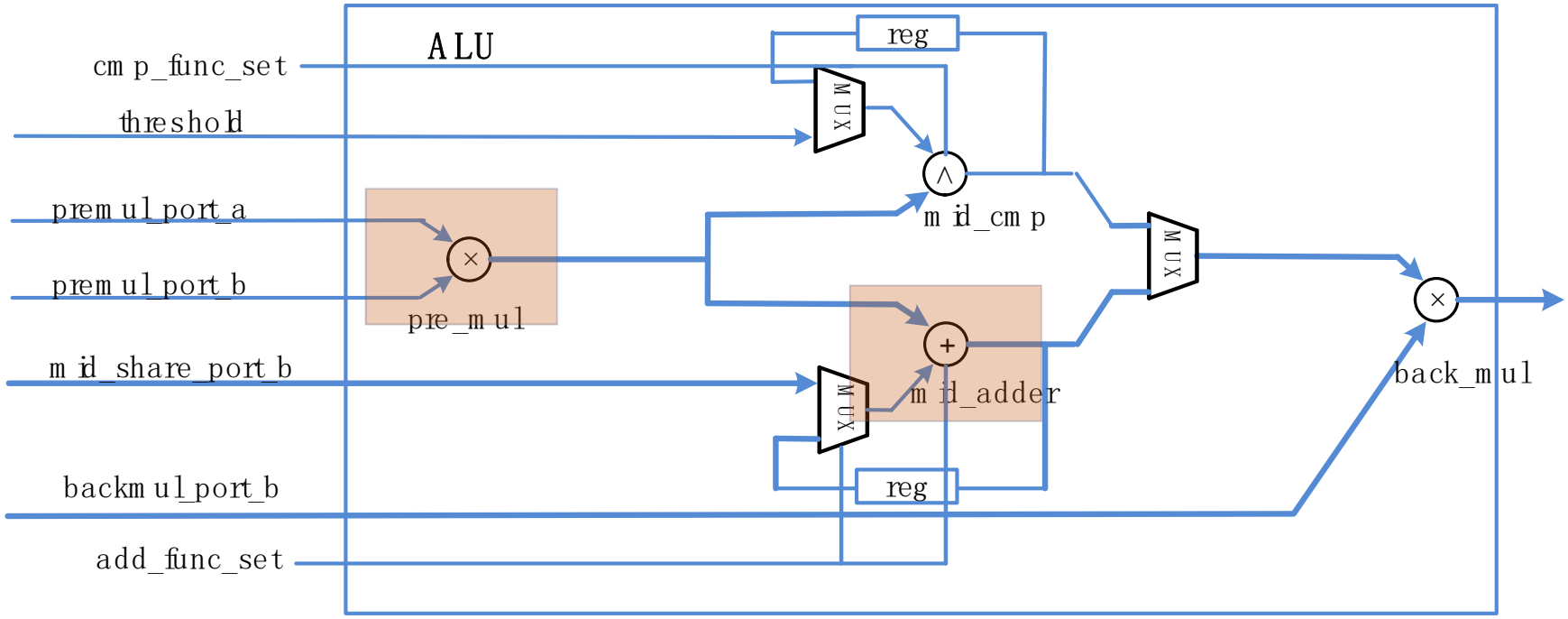
Postprocessing – FPU

- Reconfigurable ALU



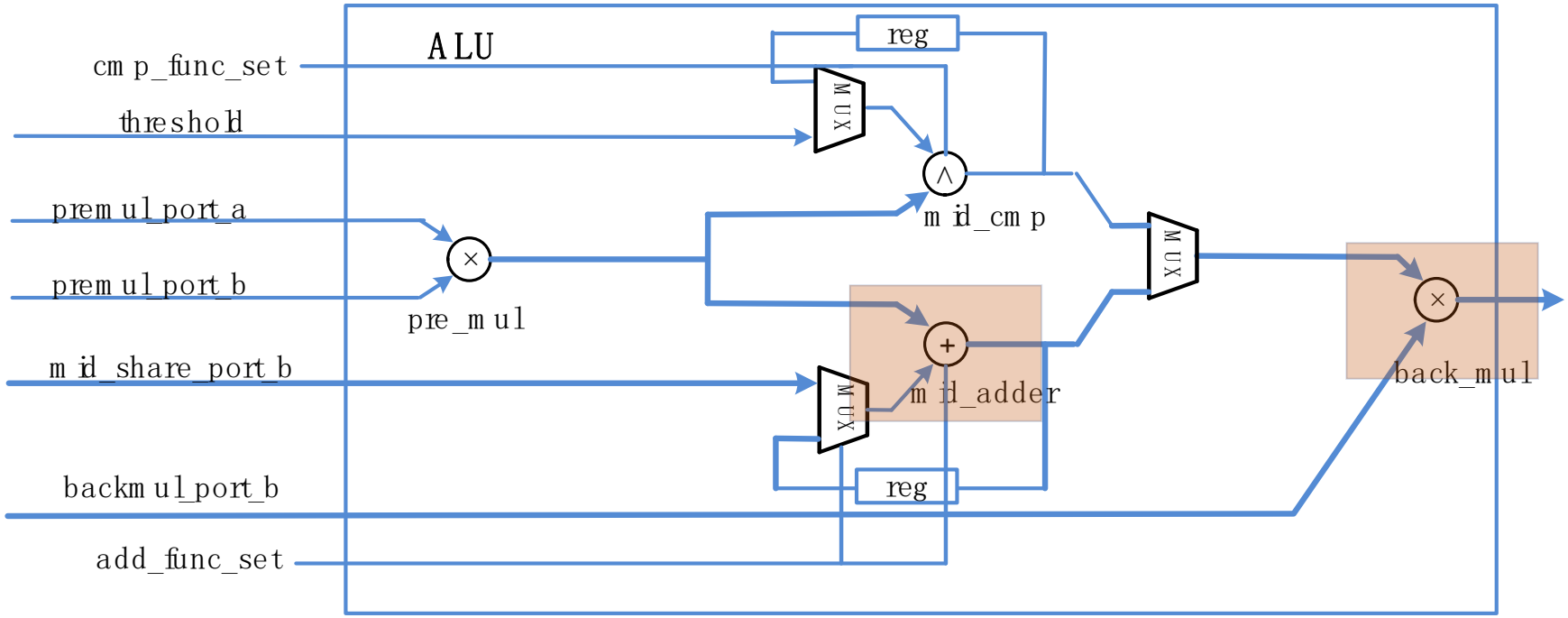
Postprocessing – FPU

Reconfigurable ALU



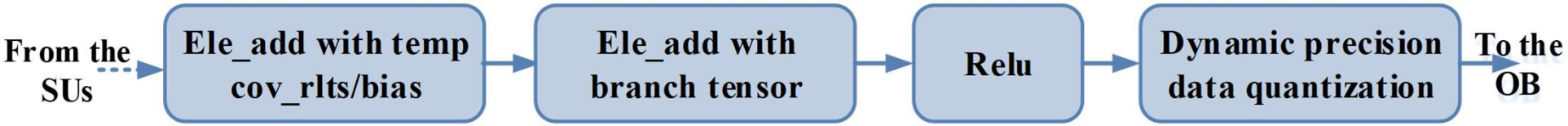
Postprocessing – FPU

Reconfigurable ALU



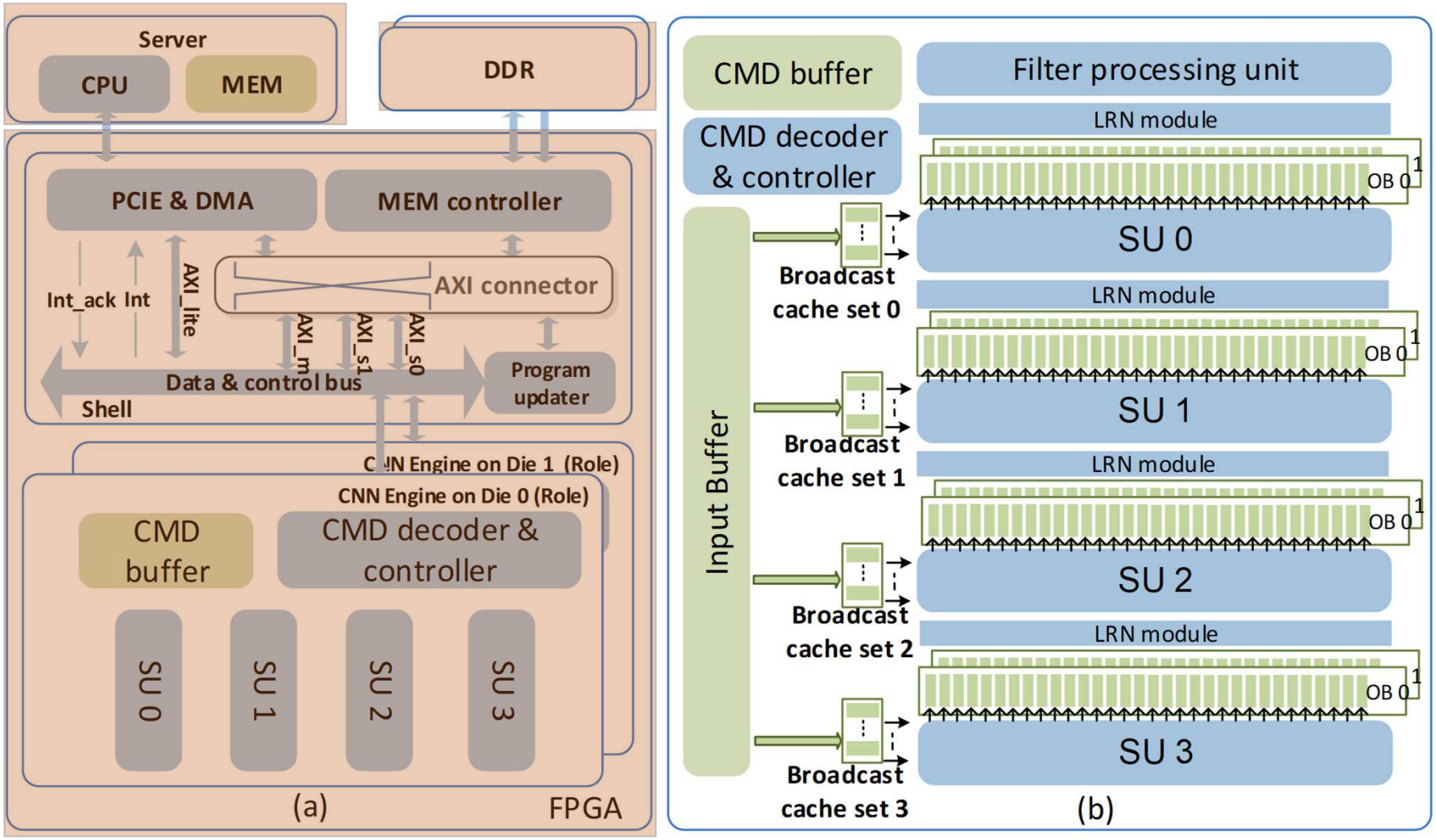
Postprocessing – Operator Fusion

- Avoid extra memory access
- Four operations fuse with Convolution

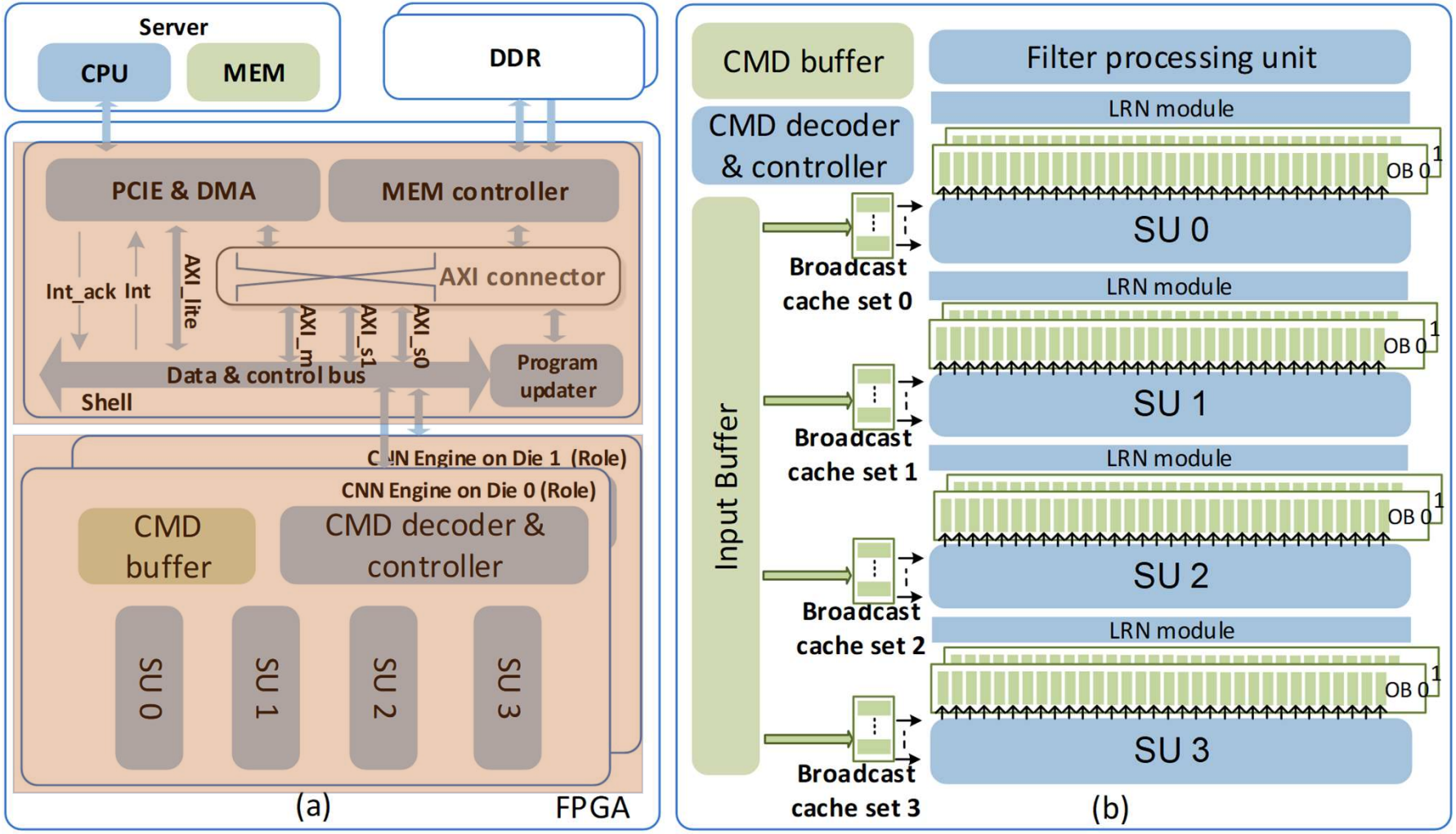


[2] J. Qiu, et al. “Going deeper with embedded FPGA platform for convolutional neural network,“ Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. ACM, 2016

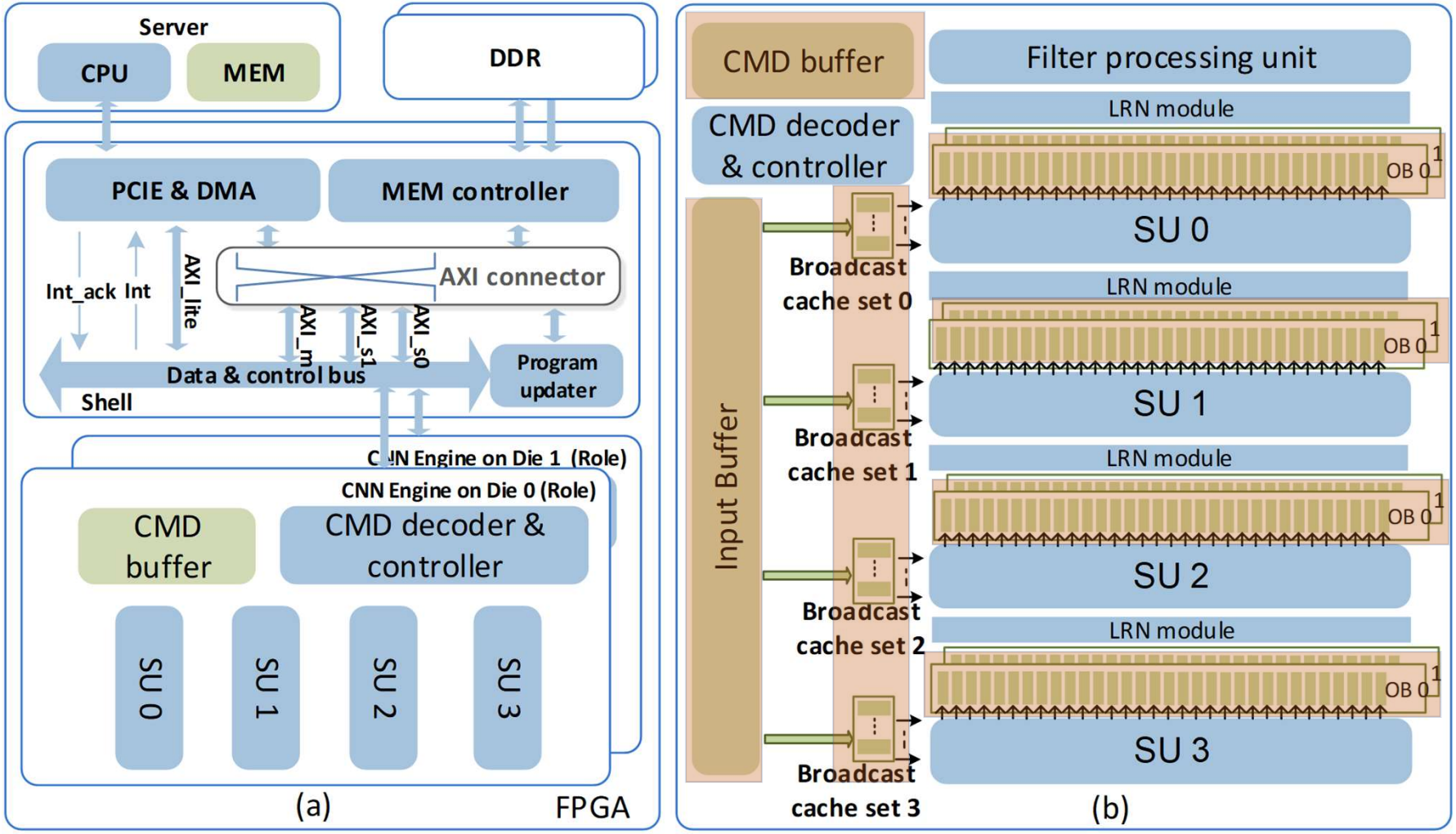
System Overview



System Overview

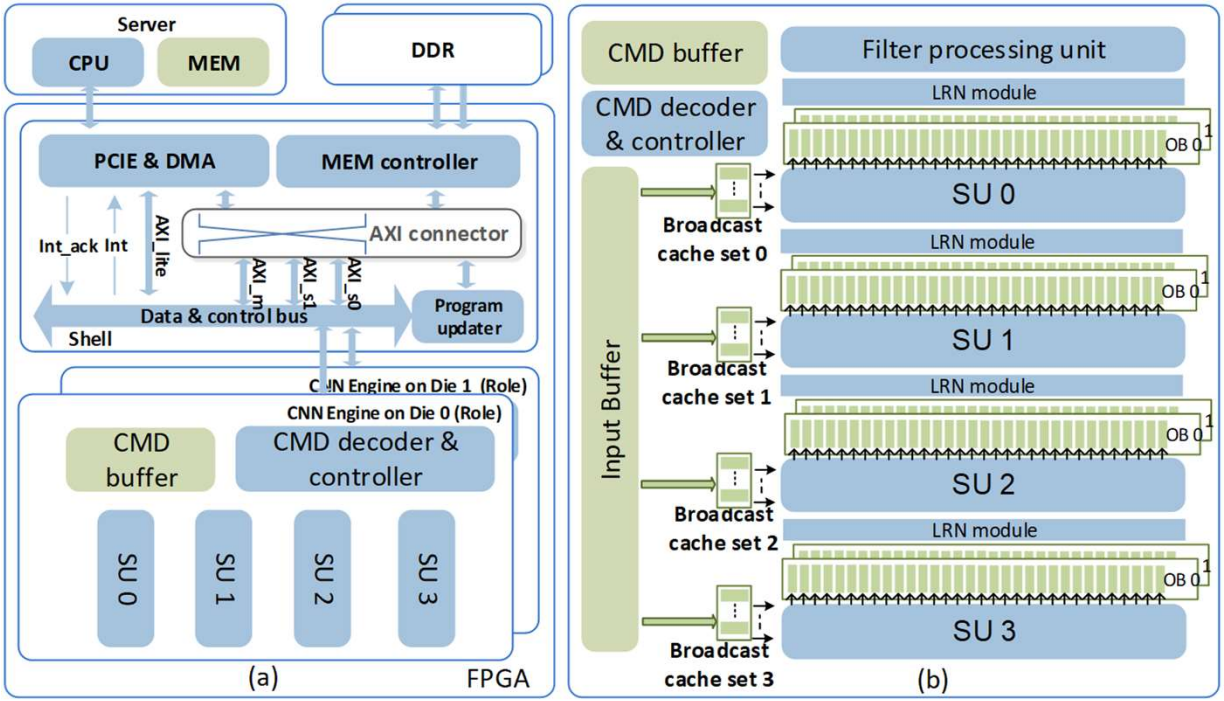


System Overview



System Overview

- Frequency
  - 500 MHz is applied for the EPEs.
  - 250 MHz is applied for the others.
- DSP
  - Each SU : 512 DSPs
  - Each CNN Engine : 4 Sus 2048 DSPs
  - Whole Chip : Two CNN Engines
  - 4096 DSPs provide 4.2 TOP/s @int16 for Conv
- Memory
  - IB : 4.2Mbit * 2
  - Bandwidth : 16 GB/s (R) -> 64 GB/s with BC
  - OB : 4.2Mbit * 2
  - Bandwidth : 64 GB/s *2 (R and W)



Experimental Results: Performance in three models

- Alexnet
- GoogLeNet
- HCNet (high-concurrency network)

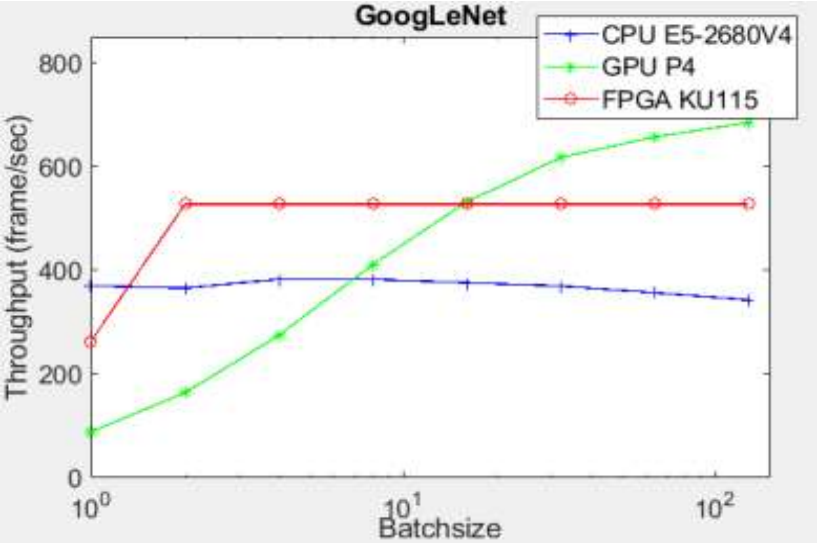
	AlexNet	GoogLeNet	HCNet
Data precision	16-bit	16-bit	16-bit
Clock (MHz)	250/500	250/500	225/450
Batch size	4	2	4
CNN size (MOPs)	1331.6/1448.8	3081.0/3083.1	444
Throughput (FPS)	1753.8	527.7	1465.1
Performance (GOP/s)	2335.4	1625.9	650.5
Latency (ms)	2.3	3.8	2.7
Power (watts)	62.6	56.6	57.6
Speedup VS P4 (7 ms)	1.4	3.9	3.4
Energy efficiency (GOP/s/W)	37.3	28.7	11.3

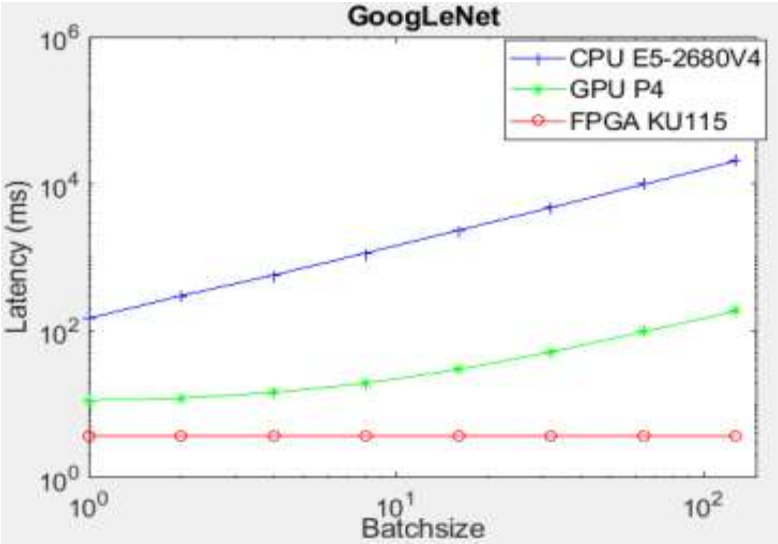
Experimental Results: Comparison with FPGA-Based Accelerators

	[3]	[4]	[5]	Ours	
FPGA chip	Arria10-1150	Virtex7-690t	KU115	KU115	KU115
Network	VGG	AlexNet	VGG	GoogLeNet	AlexNet
CNN size (GOPs)	30.8	1.4	30.8	3.1	1.3
Freq (MHz)	385	150	235	250/500	250/500
Precision	Fix16	Fix16	Fix16	Fix16	Fix16
DSPs (used/total)	2756/3036	2833/3600	4318/5520	4214/5520	4214/5520
Peak performance (TOP/s)	2.1	0.8	2.1	4.2	4.2
Real performance (TOP/s)	1.79	0.6	2	1.63	2.3

[3] J. Zhang, and J. Li. “Improving the performance of opencl-based fpga accelerator for convolutional neural network,” Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. ACM, 2017.
[4] C. Zhang C, et al. “Caffeine: towards uniformed representation and acceleration for deep convolutional neural networks,” Proceedings of the 35th International Conference on Computer-Aided Design. ACM, 2016, p. 12.
[5] X. Zhang, et al. “DNNBuilder: An automated tool for building high-performance DNN hardware accelerators for FPGAs,” Proceedings of the International Conference on Computer-Aided Design. ACM, 2018.

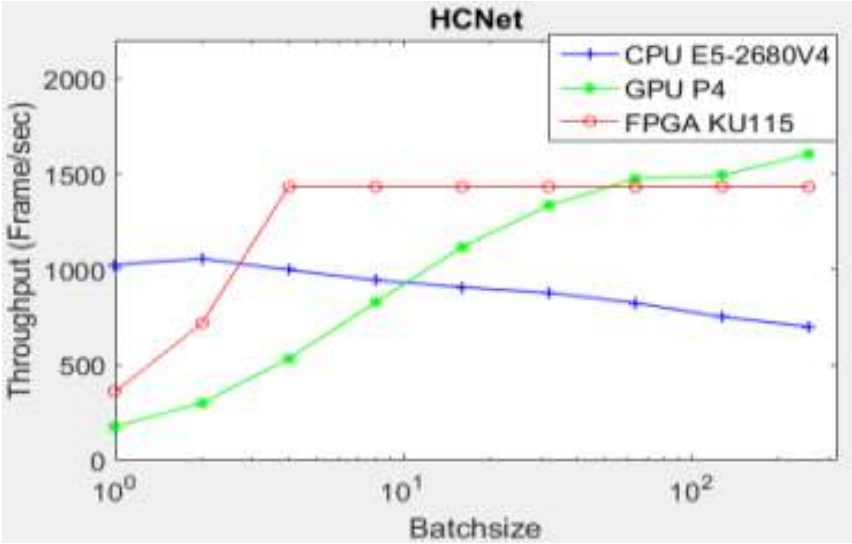
Experimental Results: Comparison with CPUs and GPU in datacenter

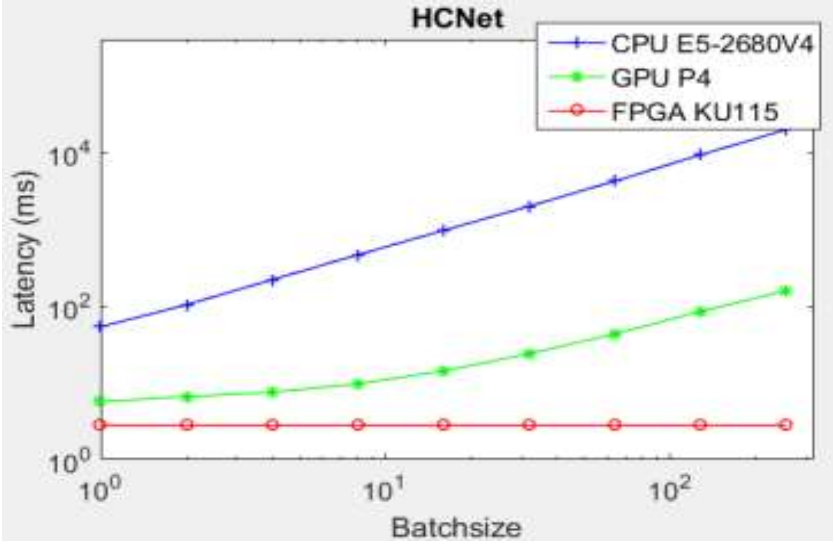




Processor	Processor per server	TOP/s		nm	MHz	On-chip memory (MB)	Off-chip memory BW (GB/s)	Power (Watts)	Release
		16bit	FP32						
Intel E5-2680V4	2	-	-	14	2400	35 × 2	76.8 × 2	120	2016 Q1
NVIDIA P4	1	-	5.5	16	1000	10 [38]	192	50-75	2016 Q3
Xilinx KU115	1	4.2	-	20	250/500	11.8	38.4	50-66	2014 Q4

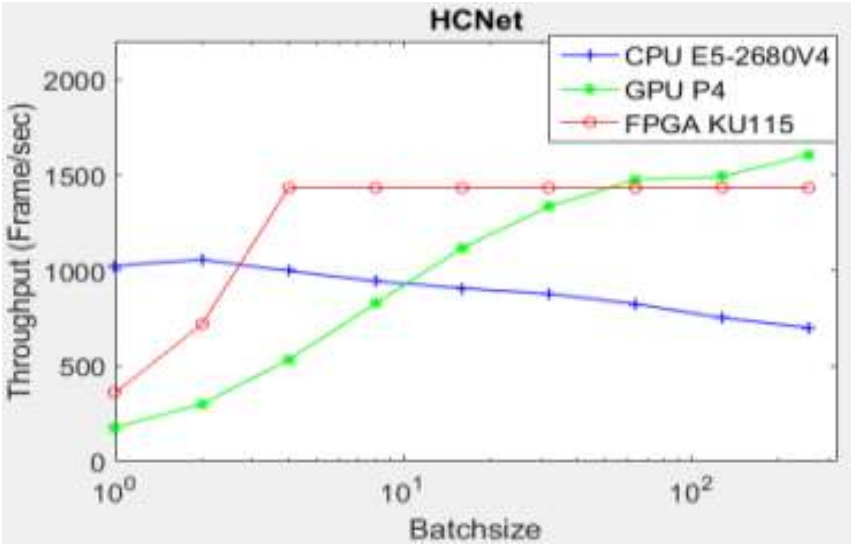
Comparison with CPU and GPU

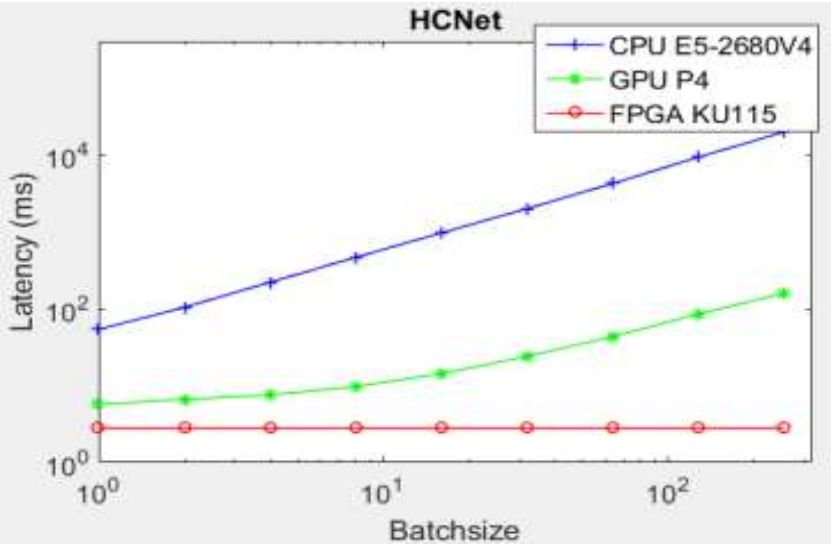




Processor	Processor per server	TOP/s		nm	MHz	On-chip memory (MB)	Off-chip memory BW (GB/s)	Power (Watts)	Release
		16bit	FP32						
Intel E5-2680V4	2	-	-	14	2400	35 × 2	76.8 × 2	120	2016 Q1
NVIDIA P4	1	-	5.5	16	1000	10 [38]	192	50-75	2016 Q3
Xilinx KU115	1	4.2	-	20	250/500	11.8	38.4	50-66	2014 Q4

Comparison with CPU and GPU





Limitations:

- simpler fabrication process
- 20% memory bandwidth
- 1/4 frequency of P4

Achievements:

- Superior performance in latency-sensitive test
- 89% throughput with 1/57 latency in throughput-sensitive test.
- Performance can be improved (UltraScale+ VU9P 16 nm)

Conclusion

- A unified framework facing different CNN models and easy to try.
- Supertile EPEs are scaled up and shaped as multiple SUs with interleaved-task-dispatching method to break computation bound
- Overcome the bandwidth limitation with dispatching-assembling buffering model and BC
- A configurable FPU is proposed to support different types of non-convolution operators

Performance	Latency	TCO	Application
4.2Top/s	50× lower	149% vs 32%	1 billion
in fix16	than GPU	than CPU	People everyday

Thank you!

Contact:
kevinxiaoyu@tencent.com