# When Dataflows Converge: Reconfigurable and Approximate Computing for Emerging Neural Networks

Di Wu and Joshua San Miguel
di.wu@ece.wisc.edu and jsanmiguel@wisc.edu

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN–MADISON

# Outline

- ❑ Motivation
- ❑ Background
- ❑ Architecture
- ❑ Evaluation
- ❑ Conclusion

# Nonlinear operations are underrated

- Emerging deep neural networks have nonlinear operations with
  - High diversity: coexistence of multiple operations

| Emerging NN | Nonlinear Operations |
|---|---|
| Capsule Neural Network (CapsNet) | *div, exp, log, sigmoid, softmax* |
| Graph Neural Network (GNN) | *div, exp, log, softmax* |
| Neural Machine Translation (NMT) | *log, sigmoid, softmax, tanh* |

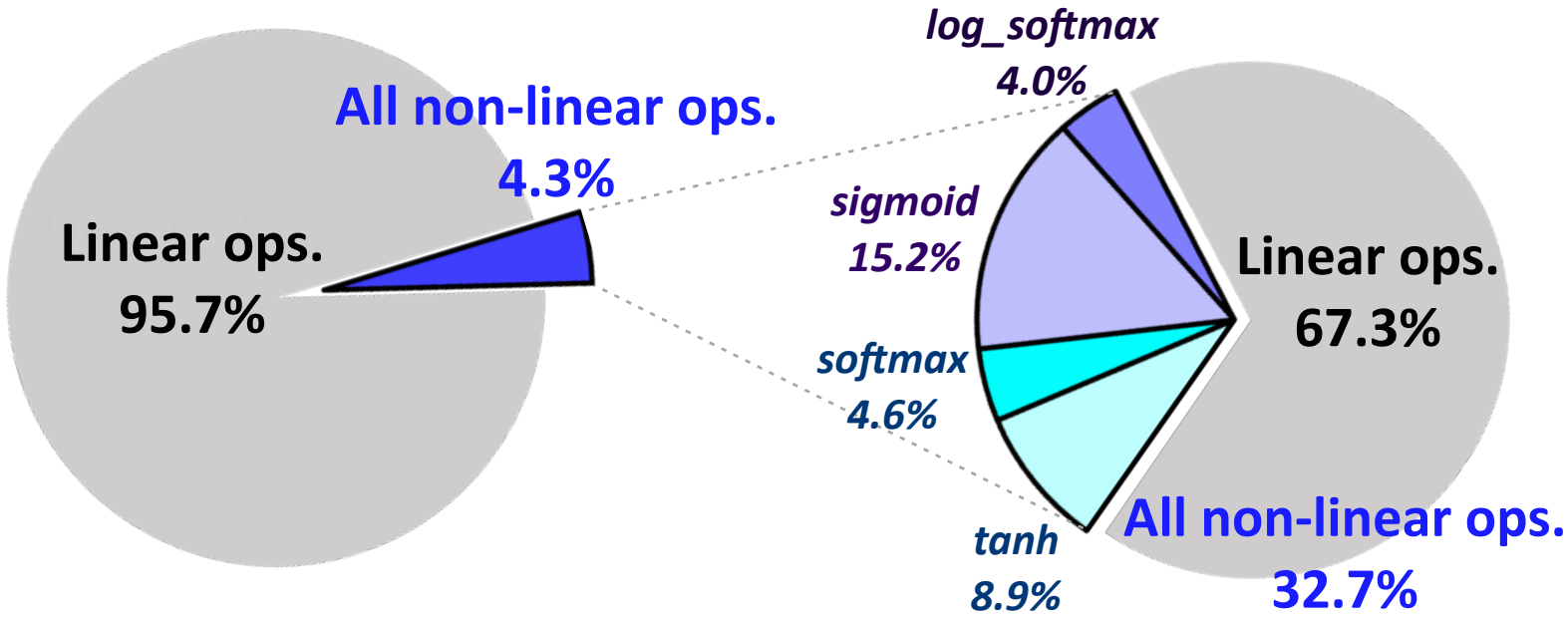


"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Nonlinear operations are underrated

- Emerging deep neural networks have nonlinear operations with
  - High diversity
  - High complexity: beyond simple ReLU and max pooling




**Operation count** **Operation runtime**

**Profiling NMT model on CPU**

"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21



Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Nonlinear operations are underrated

- Emerging deep neural networks have nonlinear operations with
  - High diversity
  - High complexity
  - High cost: costly naive processing element (PE)

| Function | Performance | Naive PE | MAC PE | Overhead |
|---|---|---|---|---|
| *MAC, div, exp, log* | Area ($\mu m^2$) | 9,323 | 1,559 | 6.0X |
| | Power (mW) | 2.35 | 0.56 | 4.2X |



"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Nonlinear operations are underrated

- Emerging deep neural networks have nonlinear operations with
  - High diversity
  - High complexity
  - High cost: costly naive processing element (PE)

| Function | Performance | Naive PE | Optimized PE | Overhead |
|---|---|---|---|---|
| *MAC, div, exp, log* | Area ($\mu m^2$) | 9,323 | 4,221 | 2.2X |
| | Power (mW) | 2.35 | 0.93 | 2.5X |

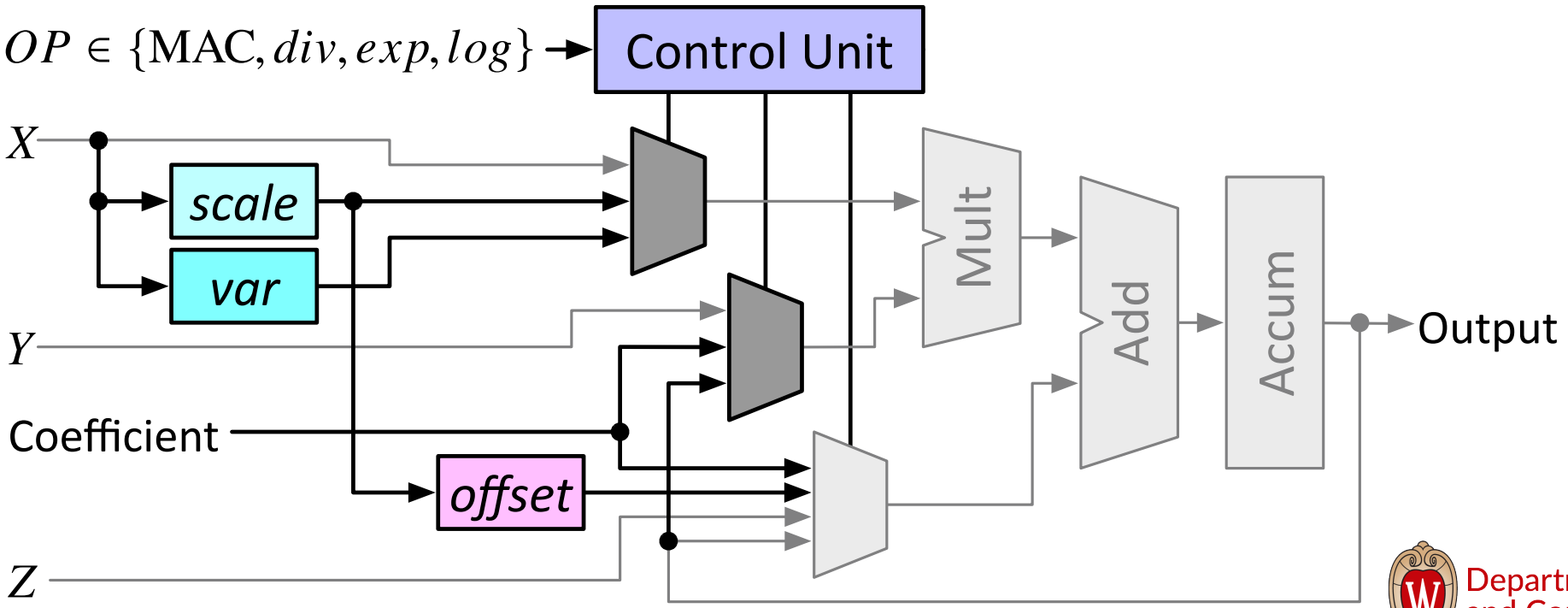


"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Unify nonlinear operation (UNO)

- Minimized overhead
  - Gray blocks: original components for MAC
  - Color blocks: extended components for nonlinear operations



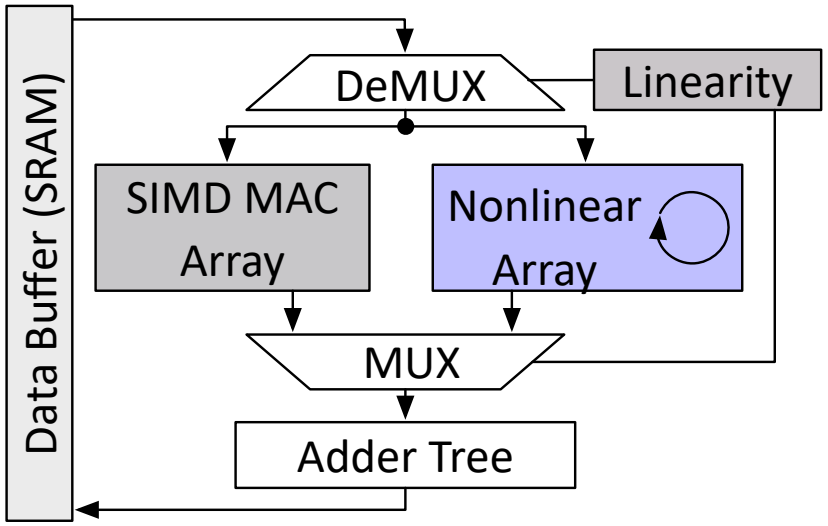


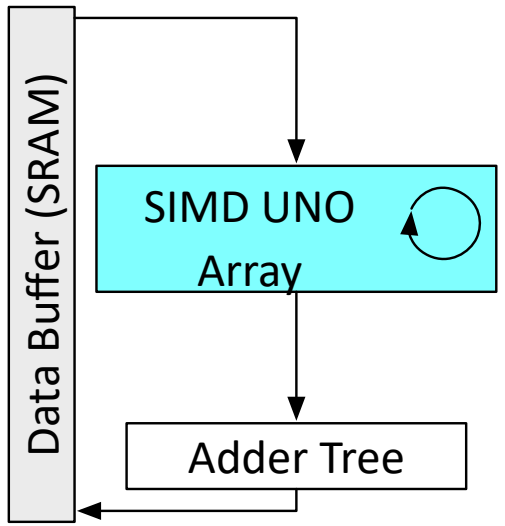

"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# UNO SIMD array

- UNO SIMD array-based accelerators
  - Naive design (left): SIMD MAC array and SIMD nonlinear array






"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Evaluate emerging neural networks

- UNO SIMD array-based accelerators
  - At least triples the energy efficiency

| | Model | Naive design | UNO SIMD | Increase (%) |
|---|---|---|---|---|
| **Area ($mm^2$)** | - | 0.659 | 0.283 | **-57.0** |
| **Power (mW)** | - | 205.5 | 66.5 | **-67.6** |
| **Throughput (samples/s)** | CapsNet | 470.3 | 567.1 | **+20.6** |
| | GNN | 6.0 | 6.0 | **+0.0** |
| | NMT | 132.5 | 160.6 | **+21.2** |
| **Energy Efficiency (samples/s)** | CapsNet | 2288 | 8527 | **+272.7** |
| | GNN | 29 | 91 | **+209.1** |
| | NMT | 645 | 2415 | **+274.5** |

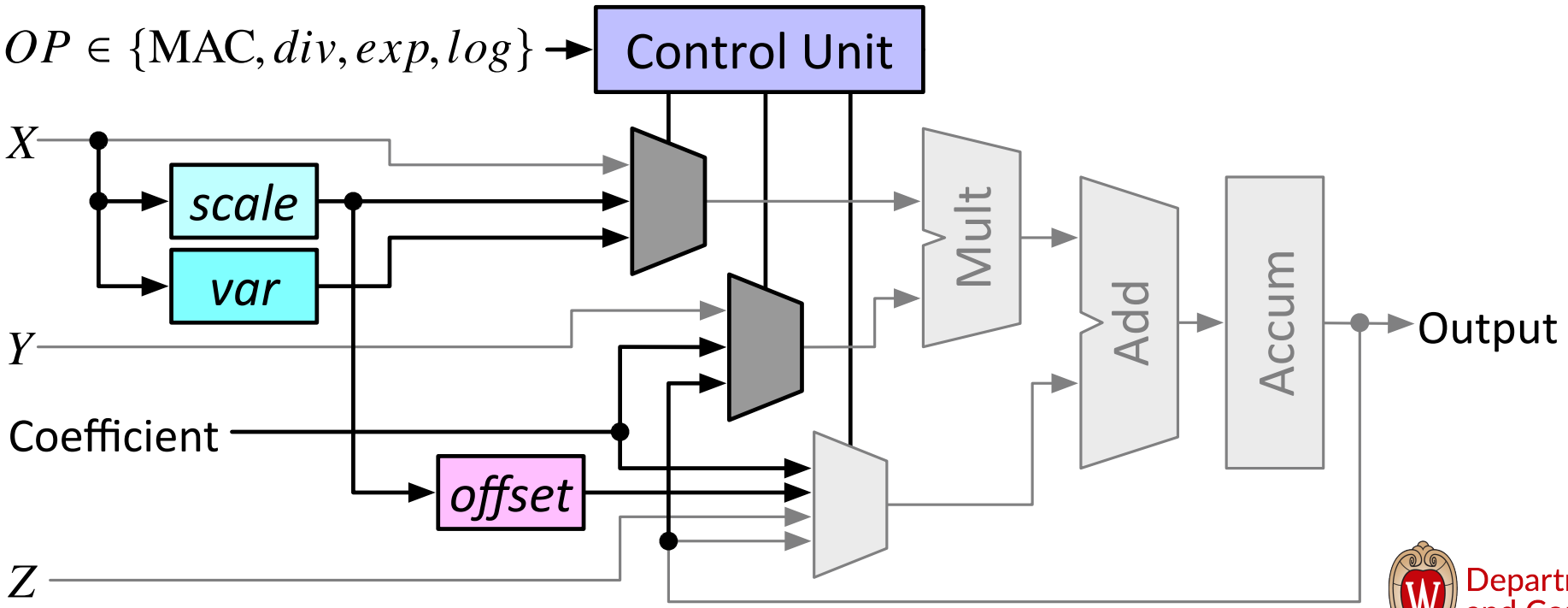


"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Evaluate emerging neural networks

- UNO SIMD array-based accelerators
  - Ancillary logic exists in every PE
  - SIMD array suffers large overall overhead






"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# UNO systolic array as a salvage

- Two dataflows converge inside one architecture
  - Inherited GEMM dataflow and extended UNO dataflow
- PE overheads disperse in PE array for extra saving
  - Separation of ancillary logic into small parts
- Conventional approximation techniques are compatible
  - A heuristic microarchitecture approximation



"When Dataflows Converge: Reconfigurable and Approximate Computing for Emerging Neural Networks", ICCD'21 (Invited)

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Outline

- ❑ Motivation
- ❑ Background
- ❑ Architecture
- ❑ Evaluation
- ❑ Conclusion

# Unify nonlinear operation (UNO)

- Leverage Horner's rule
  - Taylor approximation: unify the math expression

$$f_n(x) = \sum_{i=0}^{n} \frac{f^{(i)}(a)}{i!} \cdot (x-a)^i = \sum_{i=0}^{n} c_i \cdot (x-a)^i$$

  - MAC-compatible schedule: virtualize nonlinear operations using MAC from high to low degree

$$f_n(x) = offset + mac_n \cdot scale$$

$$mac_i = \begin{cases} |c_{n-i}| + mac_{i-1} \cdot var & \text{if } 1 < i \leq n, \\ |c_{n-1}| + |c_n| \cdot var & \text{if } i = 1. \end{cases}$$

  - Universal support: *div, exp, log, tanh, sigmoid, softmax, etc.*

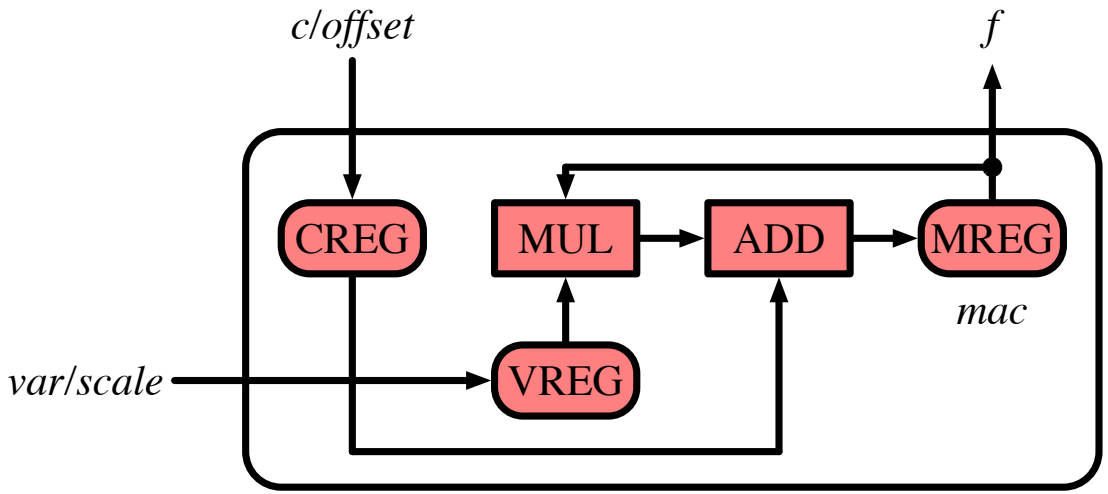


"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# UNO PE

- Weight stationary
  - Coefficient is stationary, i.e., statically stored, in CREG
  - Var and scale in VREG are calculated from input
  - Output is accumulated in MREG



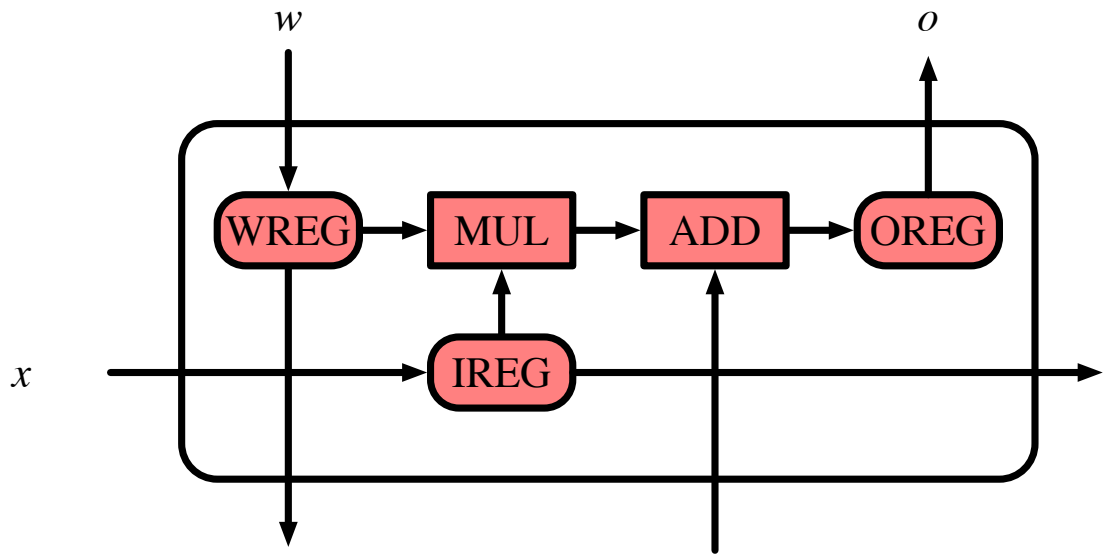



"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Systolic array PE for GEMM

- Weight stationary
  - Weight is stationary (preloaded from top to bottom)
  - Input is streamed from left to right
  - Output is streamed from bottom to top

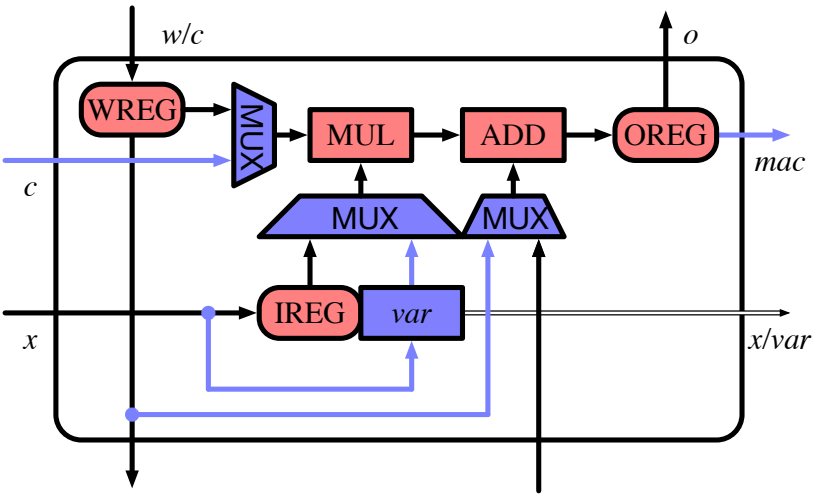


# Outline

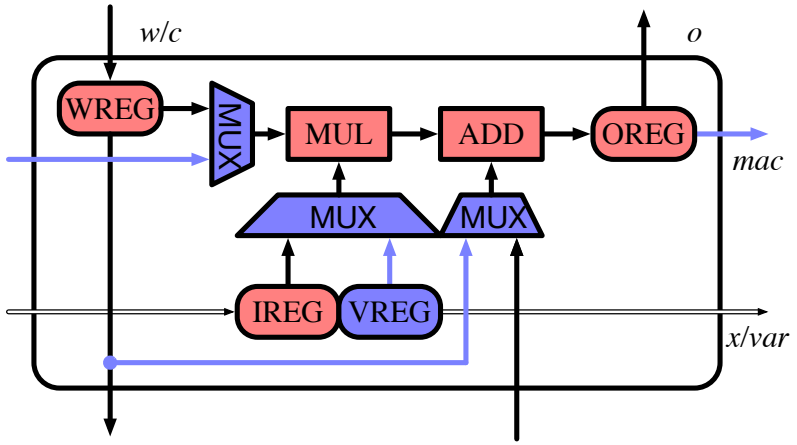
- ❑ Motivation
- ❑ Background
- ❑ Architecture
- ❑ Evaluation
- ❑ Conclusion

# UNO systolic array PE

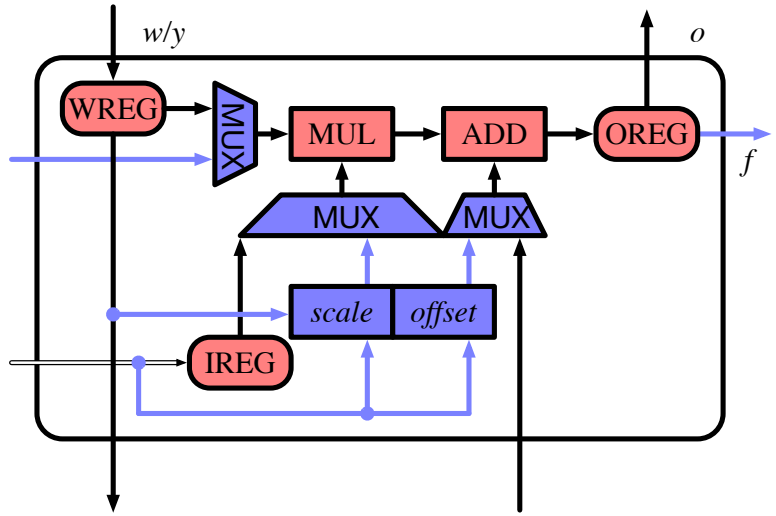
- Heterogeneous PE
  - Red block: original for GEMM dataflow
  - Blue block: extended for UNO dataflow




Leftmost column



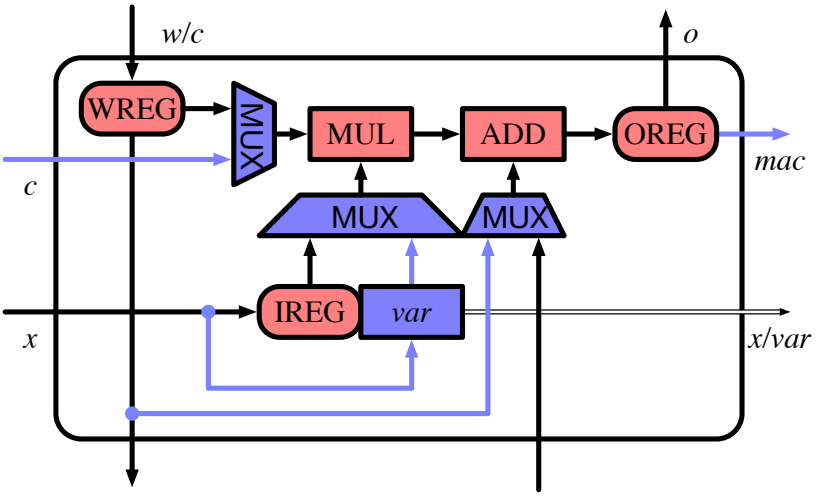

Middle column



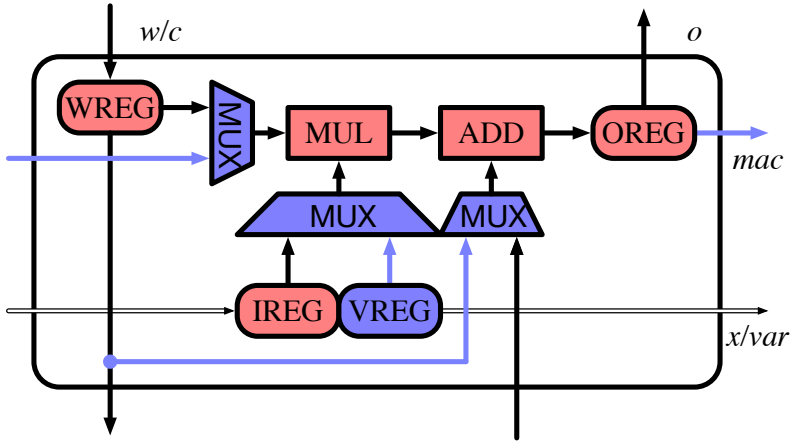

Rightmost column

# UNO systolic array PE

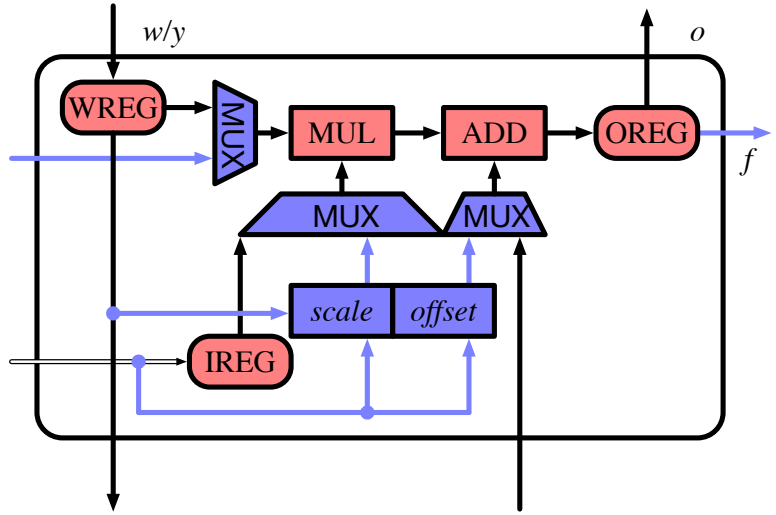
- Heterogeneous datapath
  - Black line: original for GEMM dataflow
  - Blue line: extended for UNO dataflow




Leftmost column




Middle column

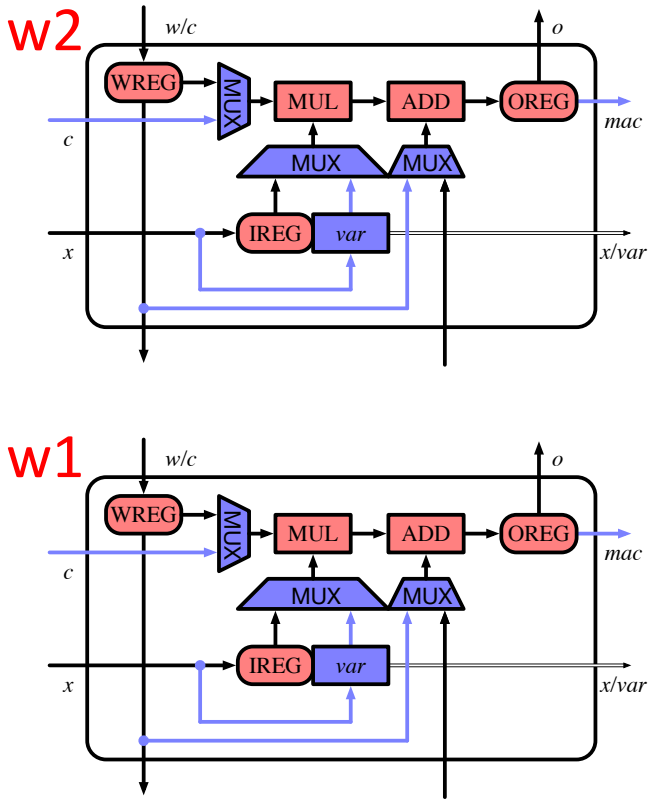



Rightmost column

# Walkthrough example-GEMM dataflow

- o=x1*w1+x2*w2
  - Each vertical column performs identical operations

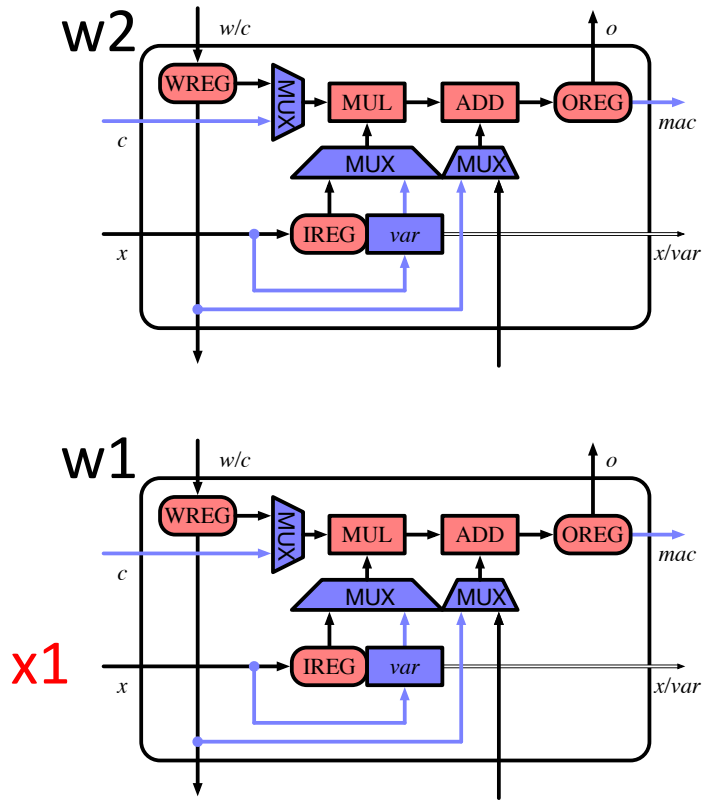
Cycle 0



# Walkthrough example-GEMM dataflow

- o=x1*w1+x2*w2
  - Each vertical column performs identical operations

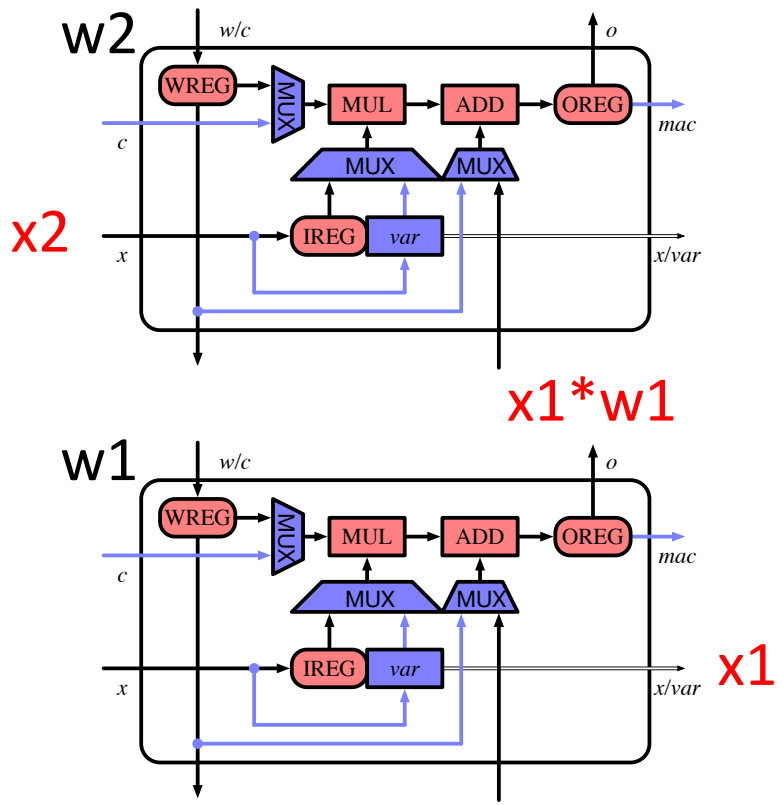
Cycle 1



# Walkthrough example-GEMM dataflow

- o=x1*w1+x2*w2
  - Each vertical column performs identical operations

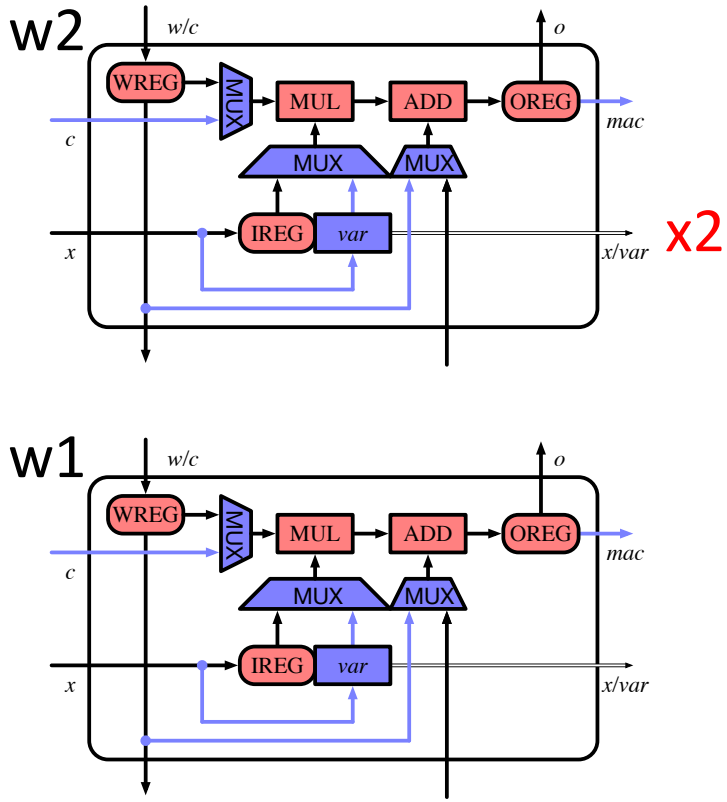
Cycle 2



# Walkthrough example-GEMM dataflow

- o=x1*w1+x2*w2
  - Each vertical column performs identical operations

o=x1*w1+x2*w2

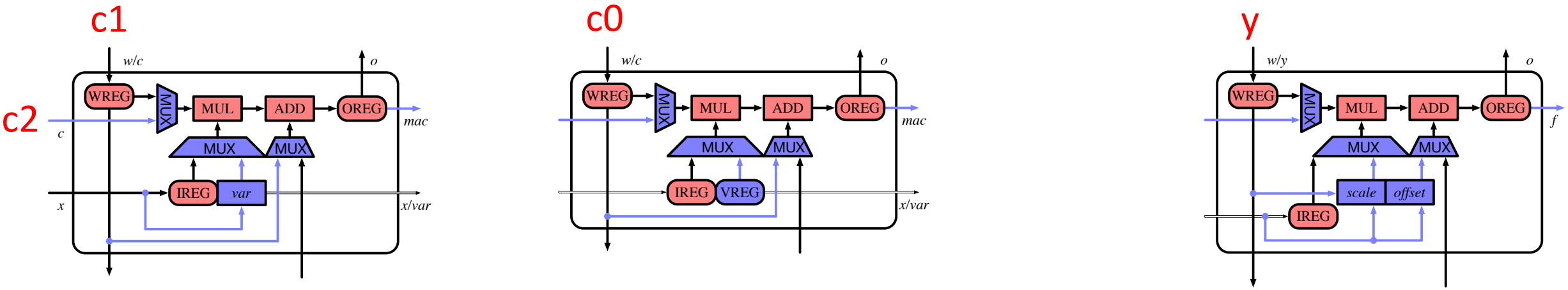



Cycle 3

**The previous MAC result is accumulated**

# Walkthrough example-UNO dataflow

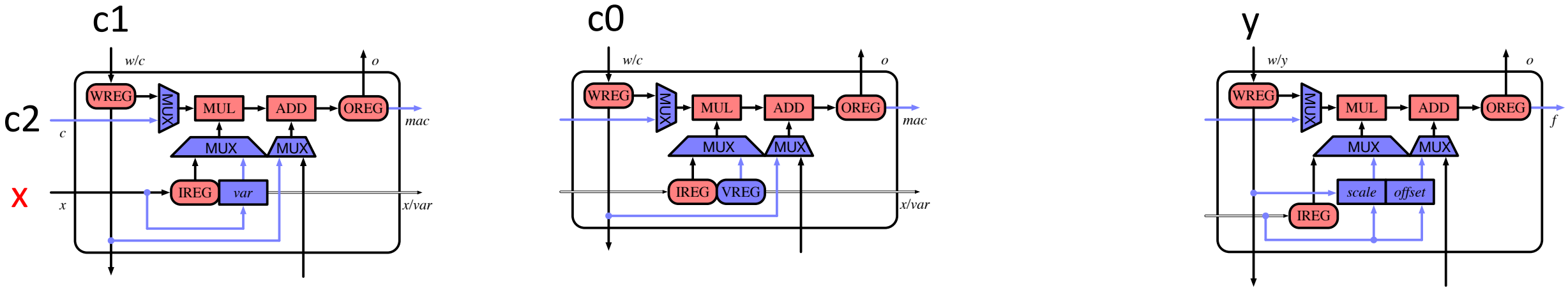
- f=(c2*var^2+c1*var+c0)*scale+offset
  - Each horizontal row performs identical operations




Cycle 0

# Walkthrough example-UNO dataflow

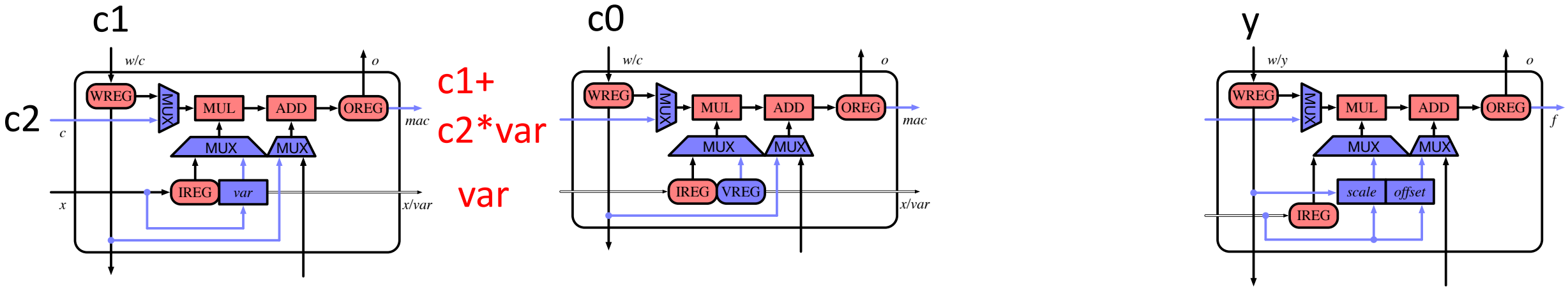
- f=(c2*var^2+c1*var+c0)*scale+offset
  - Each horizontal row performs identical operations




Cycle 1

# Walkthrough example-UNO dataflow

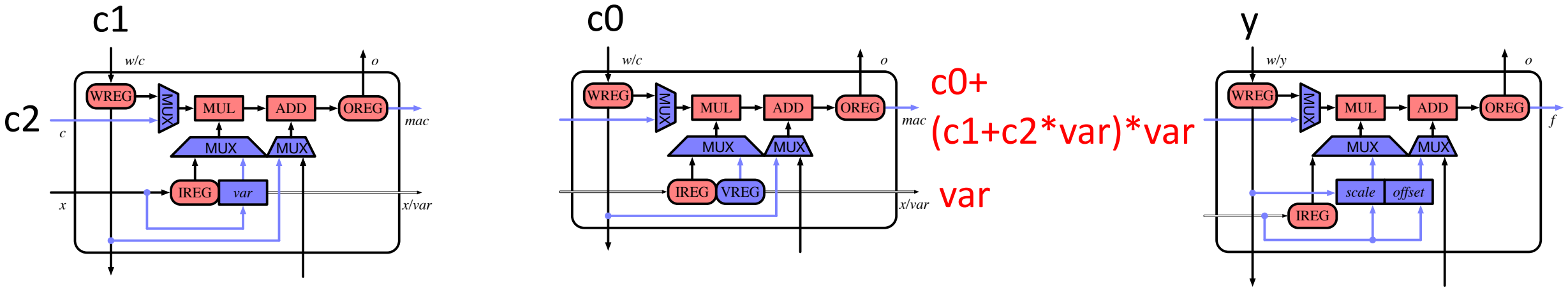
- f=(c2*var^2+c1*var+c0)*scale+offset
  - Each horizontal row performs identical operations




Cycle 2

# Walkthrough example-UNO dataflow

- f=(c2*var^2+c1*var+c0)*scale+offset
  - Each horizontal row performs identical operations




**The previous MAC result is multiplied**

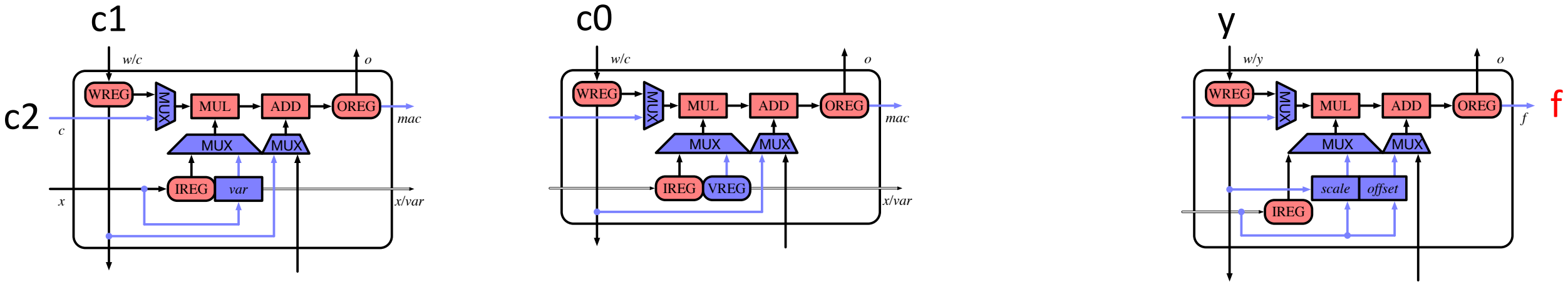
Cycle 3



Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN–MADISON

# Walkthrough example-UNO dataflow

- f=(c2*var^2+c1*var+c0)*scale+offset
  - Each horizontal row performs identical operations



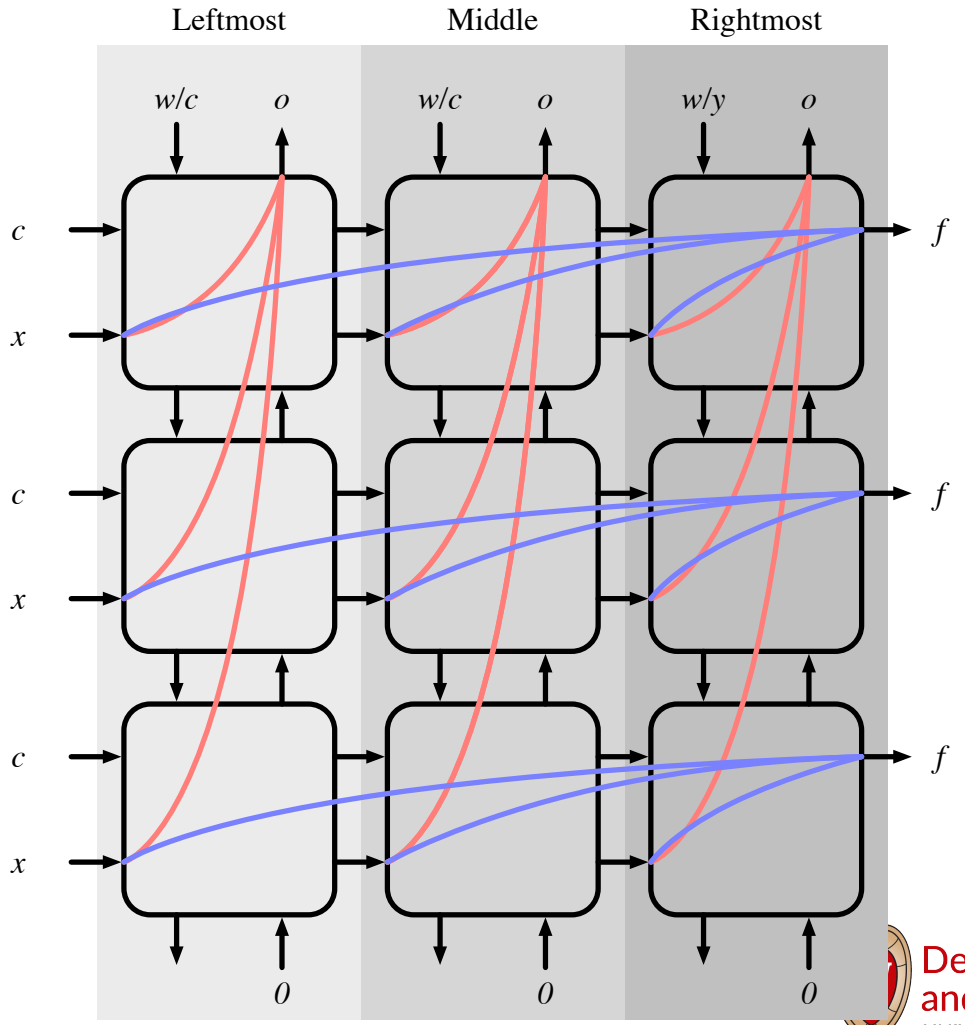

Cycle 4

# UNO systolic array

- Heterogeneous dataflows

Red line: GEMM dataflow in vertical column

Blue line: UNO dataflow in horizontal row



# Cross-level approximation

- Heuristic approximation
  - UNO algorithm proceeds from high to low degree
    - Low degree term is multiplied less, and can be more eagerly approximated
  - UNO dataflow proceeds from left to right
    - Right PE multiplies less, and can have one less bit for MAC than current PE
  - N-bit UNO systolic array of size R-by-C
    - N bits for the leftmost PE
    - (N-C+1) bits for the rightmost PE

# Outline

- ❑ Motivation
- ❑ Background
- ❑ Architecture
- ❑ Evaluation
- ❑ Conclusion

# Experimental setup

- DNN
  - CapsNet: *div, exp, log, sigmoid* and *softmax*

- Accuracy
  - Customized PyTorch UNO operator
  - Post-training quantization (no quantization-aware training)

- Hardware
  - Systolic array size R=C=4 or 8 (SIMD array size R*C=16 or 64)
  - Bitwidth N=16
  - Synthesized with TSMC 32nm technology @ 400MHz



Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN–MADISON

# Accuracy

- Operation-level
  - 16 bits: 1-bit sign, 5-bit integer and 10-bit fraction

| Operation | UNO systolic (R,C,N) | | UNO SIMD (R,C,N) | |
|---|---|---|---|---|
| | 4,4,16 | 8,8,16 | 4,4,16 | 8,8,16 |
| ***div*** | 0.022 | 0.039 | 0.021 | 0.001 |
| ***exp*** | 0.080 | 0.032 | 0.079 | 0.001 |
| ***log*** | 0.005 | 0.026 | 0.005 | 0.000 |



Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN–MADISON

# Accuracy

- Application-level
  - FP32 accuracy = 98.78%

| Sign-Int-Frac (bit) | UNO systolic (R,C,N) | | UNO SIMD (R,C,N) | |
|---|---|---|---|---|
| | 4,4,16 | 8,8,16 | 4,4,16 | 8,8,16 |
| 1-7-8 | 98.70 | 86.27 | 98.69 | 98.68 |
| 1-6-9 | 98.87 | 97.75 | 98.87 | 98.85 |
| 1-5-10 | 98.75 | 98.36 | 98.76 | 98.78 |
| 1-4-11 | 94.05 | 93.55 | 93.98 | 93.78 |

# Hardware performance

| Performance | UNO systolic (R,C,N) | | UNO SIMD (R,C,N) | | Improve (%) | |
|---|---|---|---|---|---|---|
| | 4,4,16 | 8,8,16 | 4,4,16 | 8,8,16 | 4,4,16 | 8,8,16 |
| Area ($mm^2$) | 0.13 | 0.41 | 0.18 | 0.73 | **29.4** | **43.3** |
| Power (mW) | 8.33 | 28.52 | 11.45 | 46.12 | **27.2** | **38.2** |
| Throughput (ksamples/s) | 0.15 | 0.52 | 0.15 | 0.52 | **0.0** | **0.0** |
| Energy efficiency (ksamples/J) | 18.58 | 18.34 | 13.52 | 11.34 | **37.4** | **61.7** |



Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN–MADISON

# Nonlinear operations are underrated?

- Cost of varying PEs over a MAC PE
  - Naively slabbing multiple PEs together

| Function | Performance | Naive PE | MAC PE | Overhead |
|---|---|---|---|---|
| *MAC, div, exp, log* | Area ($\mu m^2$) | 9,323 | 1,559 | 6.0X |
| | Power (mW) | 2.35 | 0.56 | 4.2X |



"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Nonlinear operations are underrated?

- Cost of varying PEs over a MAC PE
  - UNO SIMD array PE

| Function | Performance | SIMD PE | MAC PE | Overhead |
|---|---|---|---|---|
| *MAC, div, exp, log* | Area ($\mu m^2$) | 4,221 | 1,559 | 2.7X |
| | Power (mW) | 0.93 | 0.56 | 1.7X |



"UNO: Virtualizing and Unifying Nonlinear Operations for Emerging Neural Networks", ISLPED'21

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN-MADISON

# Nonlinear operations are not underrated

- Cost of varying PEs over a MAC PE
  - UNO systolic array PE (This work)

| Function | Performance | Systolic PE | MAC PE | Overhead |
|---|---|---|---|---|
| *MAC, div, exp, log* | Area ($\mu m^2$) | 1,832 | 1,559 | 1.2X |
| | Power (mW) | 0.36 | 0.56 | 0.6X |

# Outline

- ❑ Motivation
- ❑ Background
- ❑ Architecture
- ❑ Evaluation
- ❑ Conclusion

# Conclusion

- We identify the bottleneck of UNO SIMD array for emerging DNNs
  - Overhead repetition in every PE
- We propose a reconfigurable systolic array based on UNO
  - Flexibly execution of both linear and nonlinear operations
  - Cross-level approximation with heuristics
- We evaluate the proposed design on an emerging DNN
  - Tremendous performance gain without accuracy loss



Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN–MADISON

# Thank you!
# Q & A

Di Wu and Joshua San Miguel
di.wu@ece.wisc.edu and jsanmiguel@wisc.edu

Department of Electrical and Computer Engineering
UNIVERSITY OF WISCONSIN–MADISON