DRAMA: An Approximate DRAM Architecture for High-performance and Energy-efficient Deep Training System

Duy-Thanh Nguyen[1], Chang-Hong Min[1], Nhut-Minh Ho[2], Ik-Joon Chang[1]

[1]Kyung Hee University, Republic of Korea

[2]National University of Singapore, Singapore

경희대학교

KYUNG HEE UNIVERSITY

Short Bio

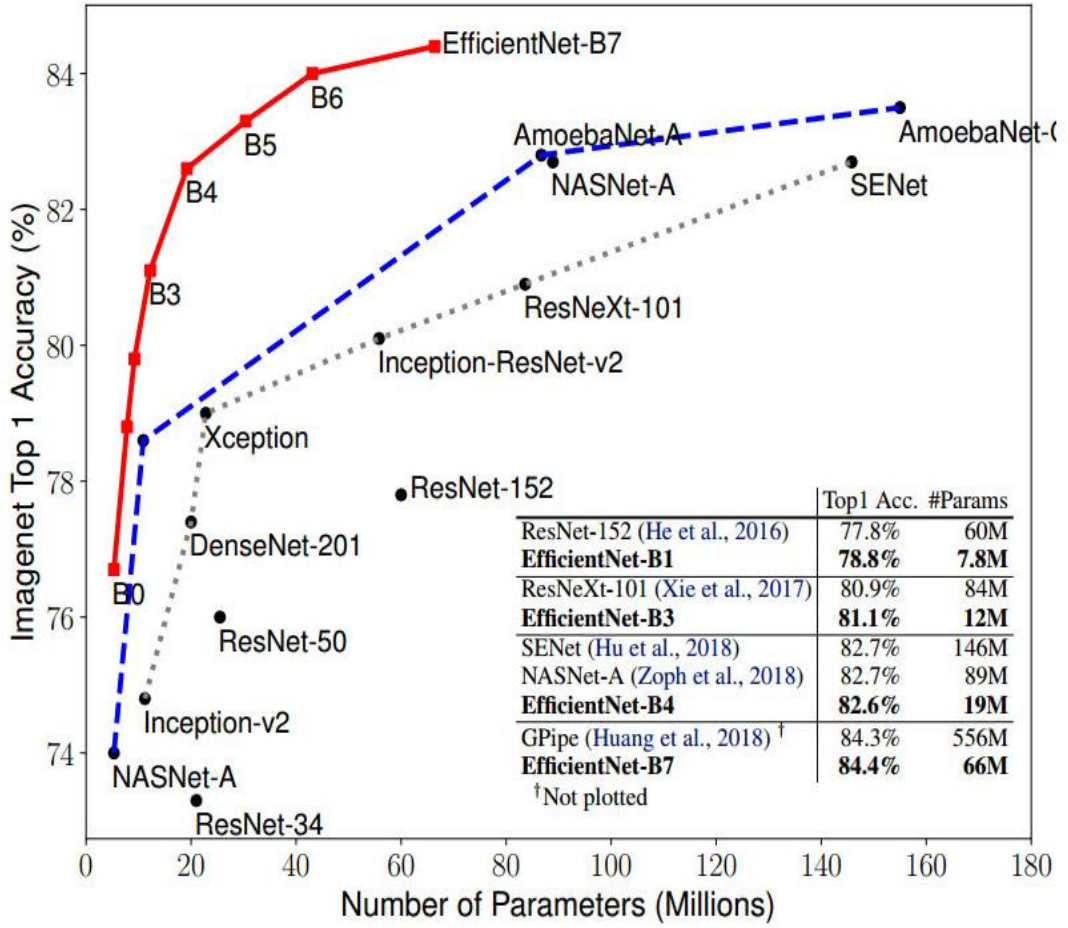
- Name: Duy-Thanh Nguyen (Ph.D Student)
- Affiliation: Kyung Hee University – Republic of Korea
- Research fields:
  - Approximate computing,
  - Energy-efficient computer architecture,
  - Reliable Memory Systems.

Outline

- Motivation
  - The important of Significant-Bit Protection in DNNs
  - The Effect of DRAM Refresh Relaxation
- Key Observations
  - Floating-point under Retention Errors
  - Large Sensitivity to Bit-errors of Some Exponent Bits
- Our Approach:
  - Energy saving: Leverage refresh/non-refresh DRAM chips
  - Performance improvement: Hide DRAM Refresh overhead in DNN application
- Validation
- Energy and Performance Simulation Results
- Conclusion

Motivation – MSBs of DNN's data are easily to get hurt

- DNNs become deeper and wider
  - The size of DNN Parameters tends to be larger
  - DRAM size should be larger
  - **DRAM power would be more significant in data-center**
- Large processing time for the training of DNNs
  - Training speed needs to be improved



	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**78.8%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.1%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**82.6%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.4%**	**66M**

†Not plotted

- Floating-points are still required for maintaining the DNN-training accuracy

  –> **Most significant bits are extremely sensitive to errors (St-DRC@DAC'2019)**

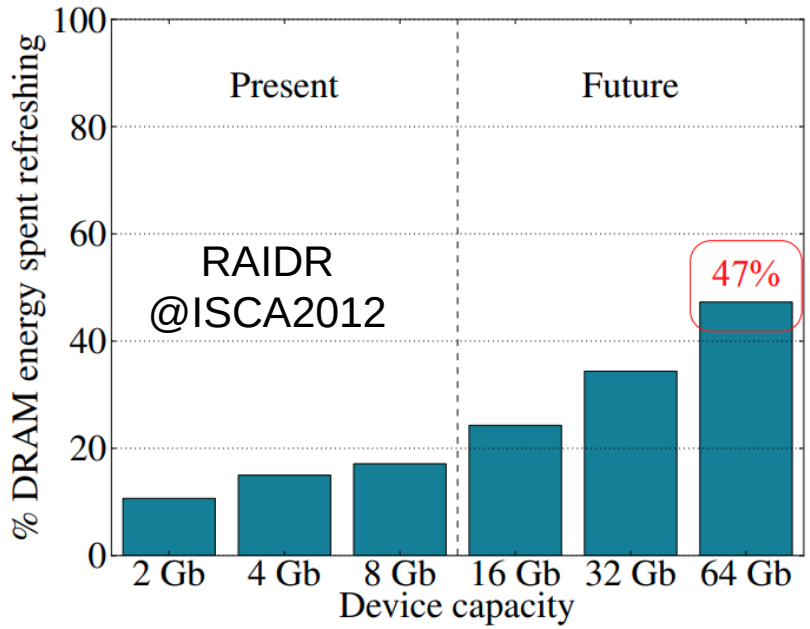
Motivation – DRAM refresh need to be eliminated

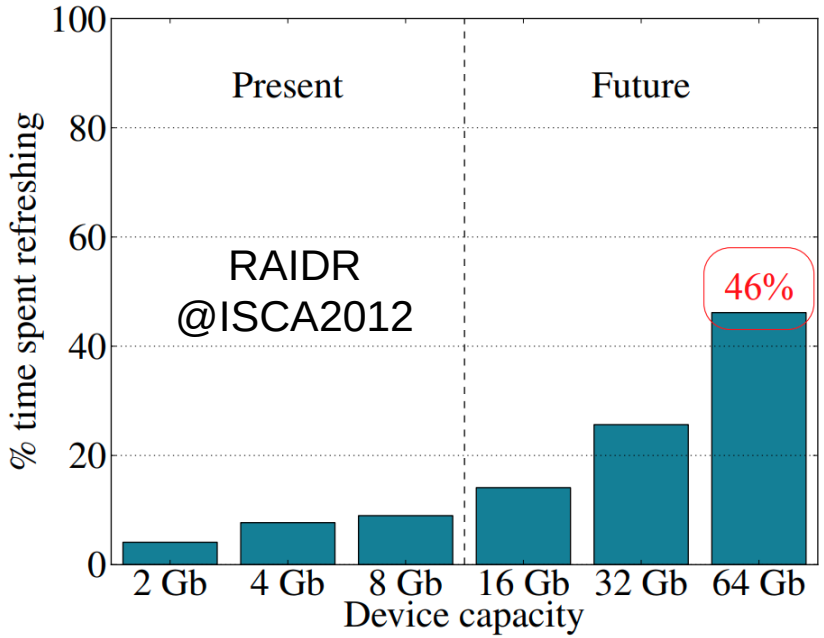
- DRAM refresh consumes up-to 47% DRAM energy *

  –> **DRAM refresh power is very significant**

- DRAM refresh take up-to 46% DRAM performance*

  –> **The refresh overhead needs to be reduced, improving the system performance**





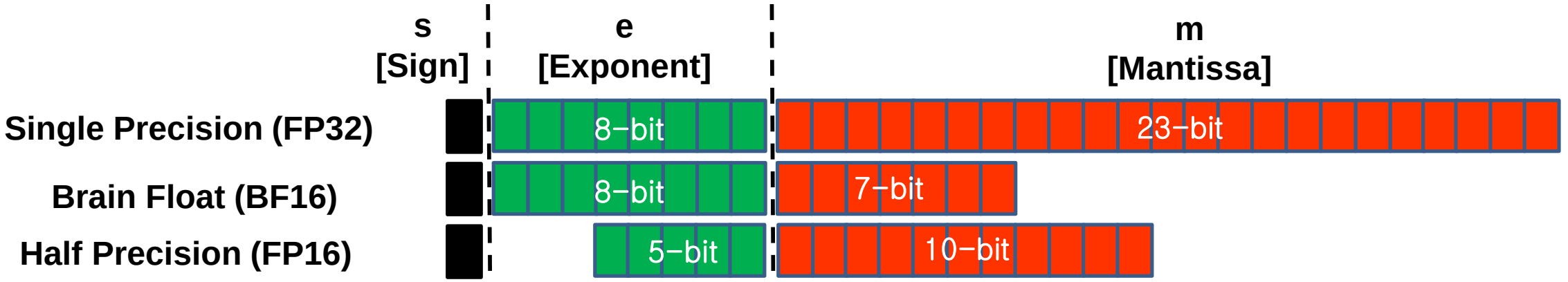
*Malcolm @HPCA'2010/Lefurgy@TCOM'2003/Ghose@SIGMETRIC'2018

SOTA works on DRAM Refresh Relaxation of DNN

	Approx.DRAM @ISCAS18	**St-DRC @DAC19**	**PCM @DATE20**
DRAM Power Saving Rate	15%	23%	22%
System Performance Improvement Rate	-10%	0.12~4%	-5%
Application	DNNs	DNNs	DNNs
Precise/Approximate	Approximate	Approximate	Approximate
Challenge	Change the cache design. The huge overhead for multiple row accesses	ECC overhead to protect the significant-bits	Change the cache design. The huge overhead for multiple row accesses

- Our Contribution: Negligible accuracy degradation in DNN in spite of some retention errors, Reasonable verification effort, Significant power saving, System performance improvement

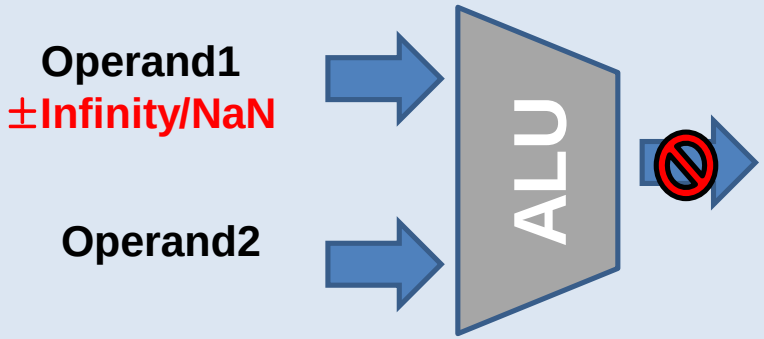
Floating point IEEE 754 under Retention Errors



- **Conversion {s,e,m} → {$-1^{s} \times M \times 2^{(e-Bias)}$ | Bias = 127 for FP32/BF16 or 15 for FP16, M = 1.m}**

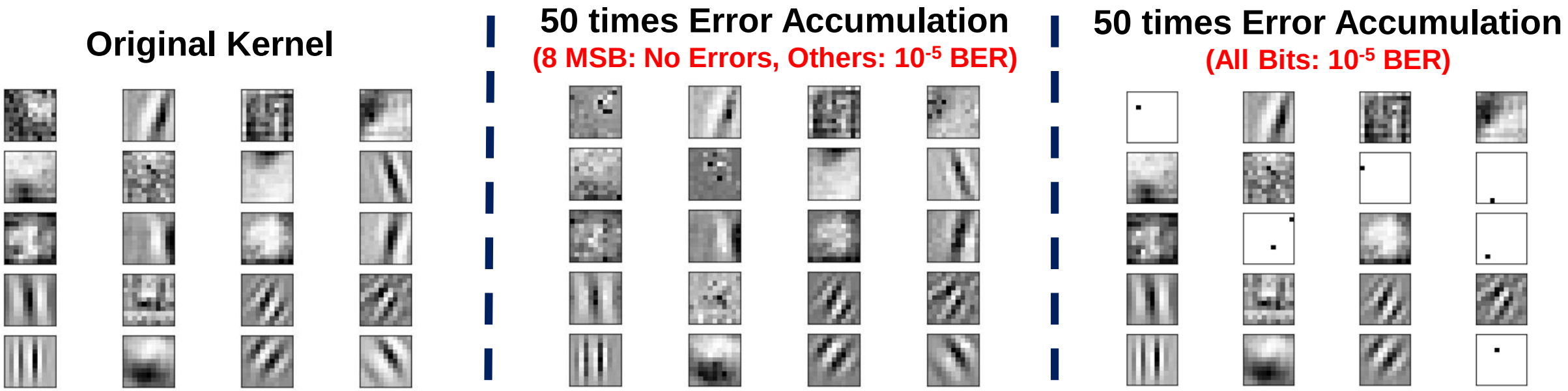
Significant Challenge Due to Retention Errors

When all 1's are in the exponent,
the data will become ±Infinity/NaN



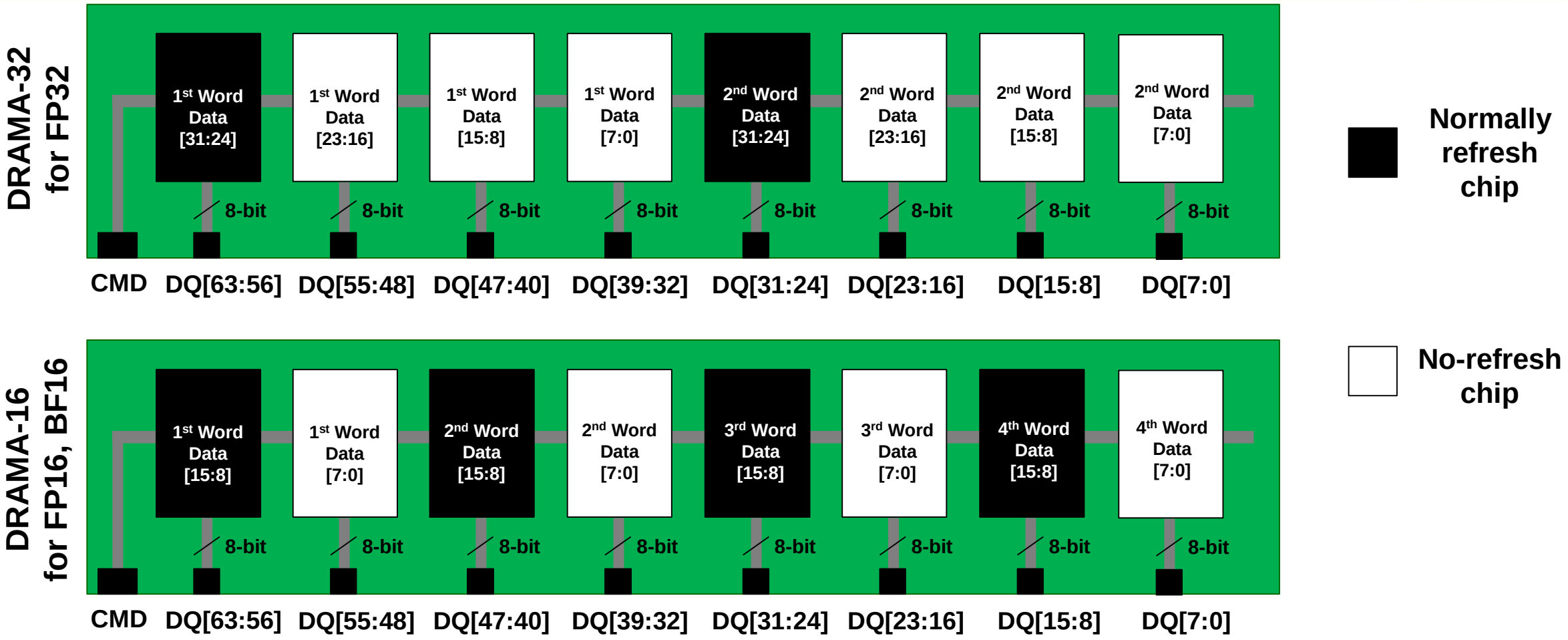
- **Error propagation due to +/- Infinity and NaN → Catastrophic system errors**

Large Sensitivity to Bit-errors of Some Exponent Bits (Inference)



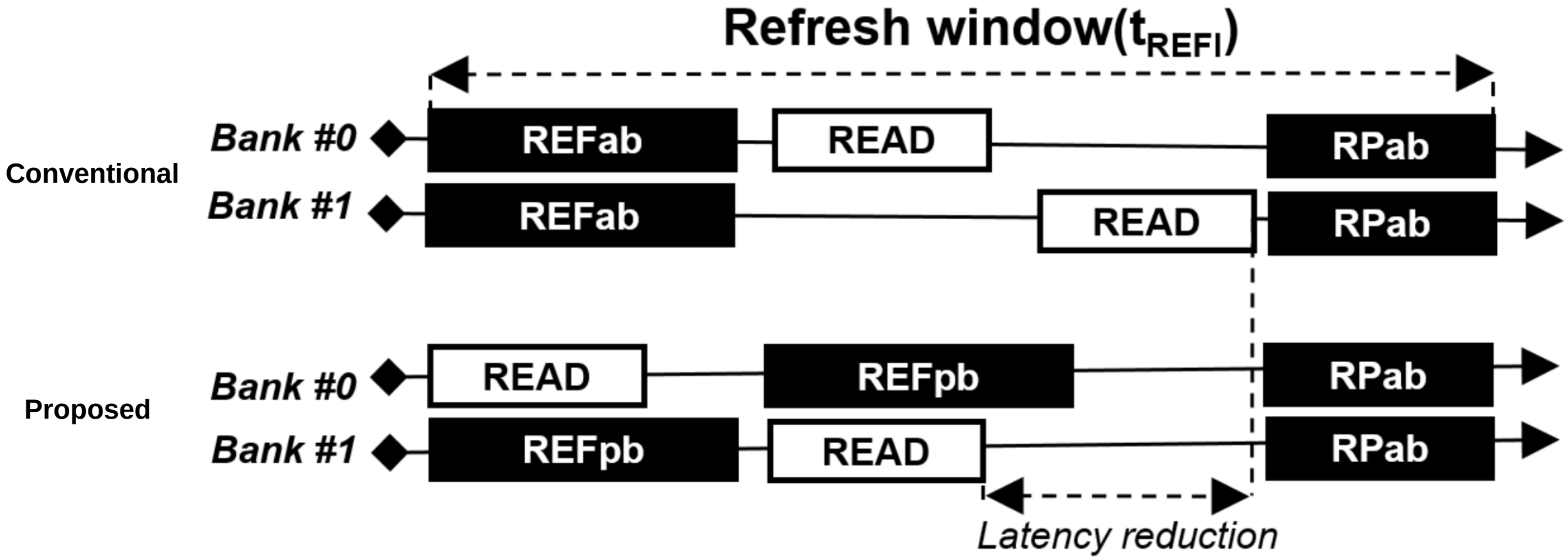
- Exponent bits are extremely sensitive to the error
- Our approaches:
  - No refresh for less-critical LSB's of DNN data
    - Reduce DRAM energy in DNN Training
  - Critical MSB's of DNN are normally refreshed, but hiding the performance overhead due to the refresh
    - Improve system performance in DNN Training

Our Approach: Data Mapping for Energy saving in DNNs



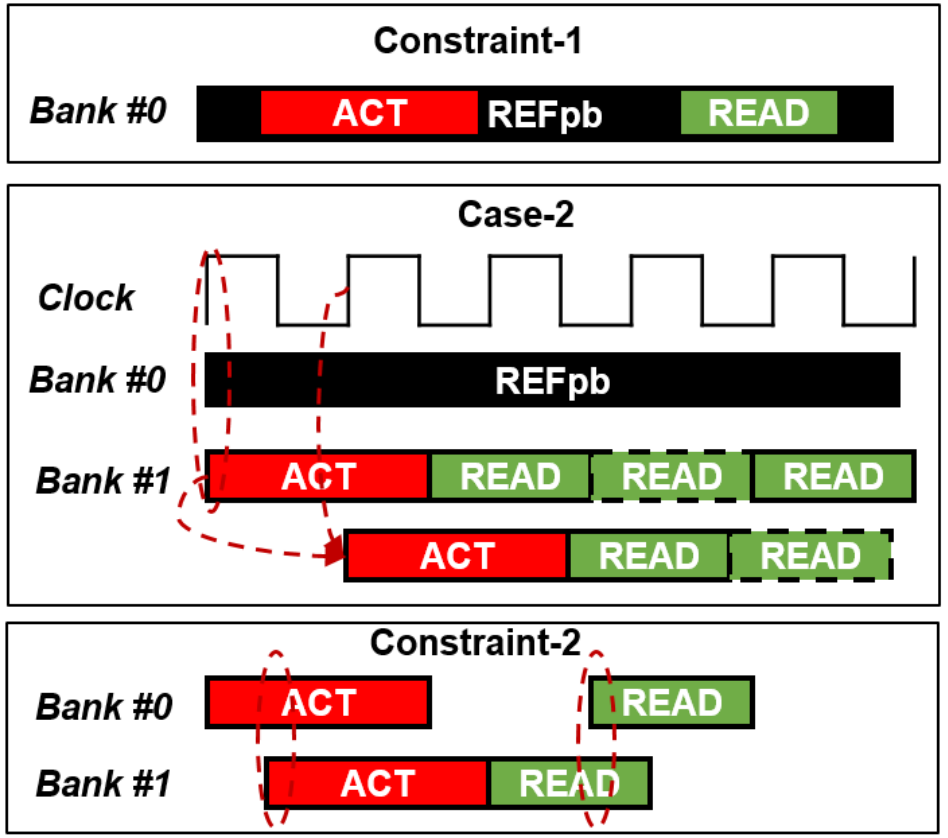
- Place the important bits in Normally refresh DRAM chips

Our Approach: Performance improvement in DNN-training

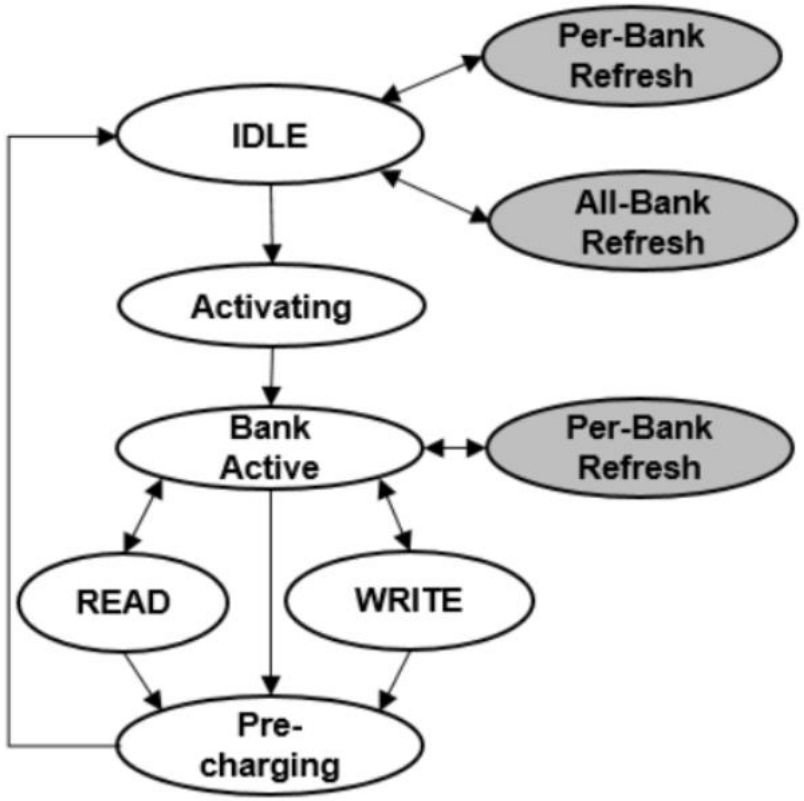


- To improve the performance → Hide the refresh overhead for the normally refreshed DRAM chips
  - –Per-bank refresh command is used.

Some Constraints for Per-bank refresh



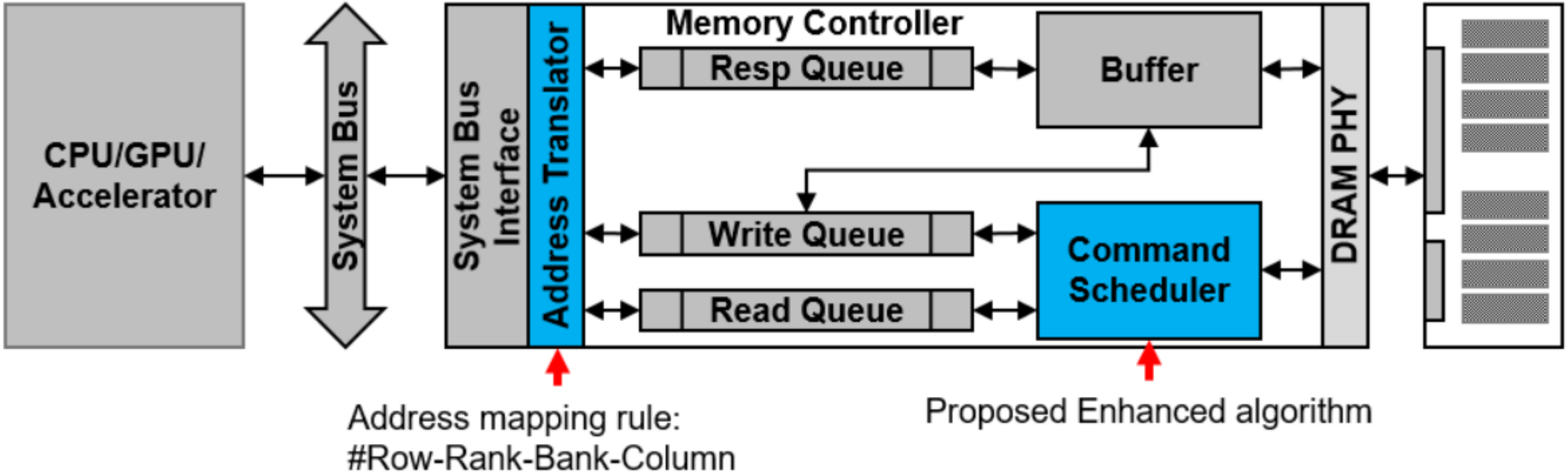
a. Paralellizing bank restriction



b. LPDDR4 state diagram

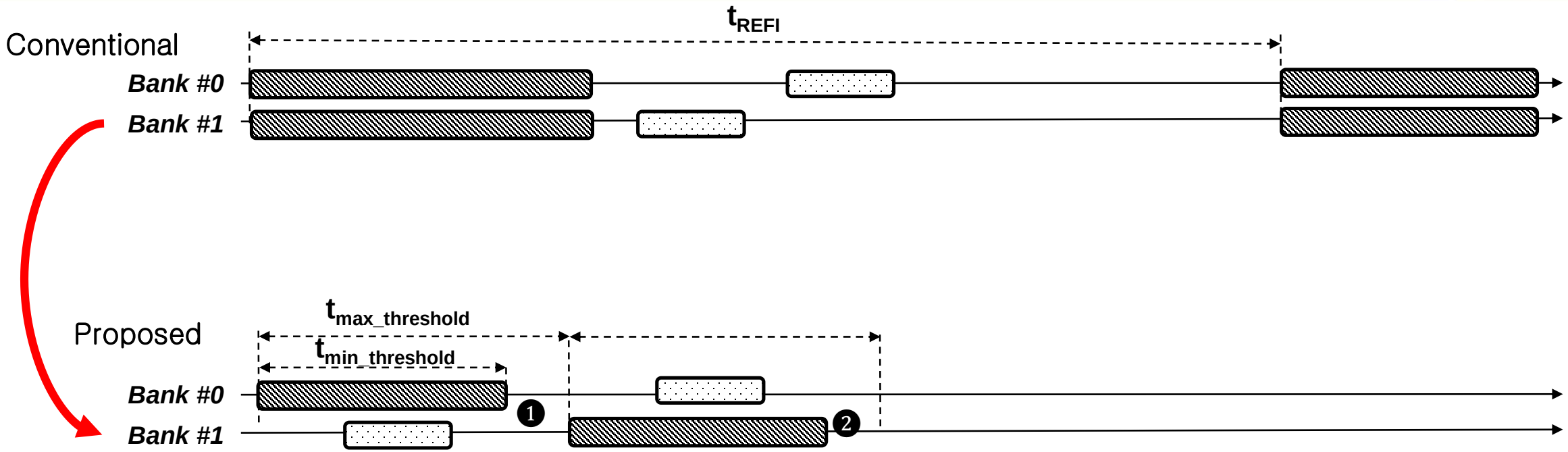
- We put some constraints to prevent possible hazards due to per-bank refreshes
  - Constraint-1: A bank under the per-bank refresh operation cannot be activated or accessed.
  - Constraint-2: When a certain bank is under the per-bank refresh operation, another activated bank can be accessed. However, except the per-bank refresh operations, other operations cannot be paralleled.

DRAMA Memory Controller



- **Two major changes in the DRAM Controller to support the per-bank refresh**
  - **Command Scheduler**
  - **Address Translator**

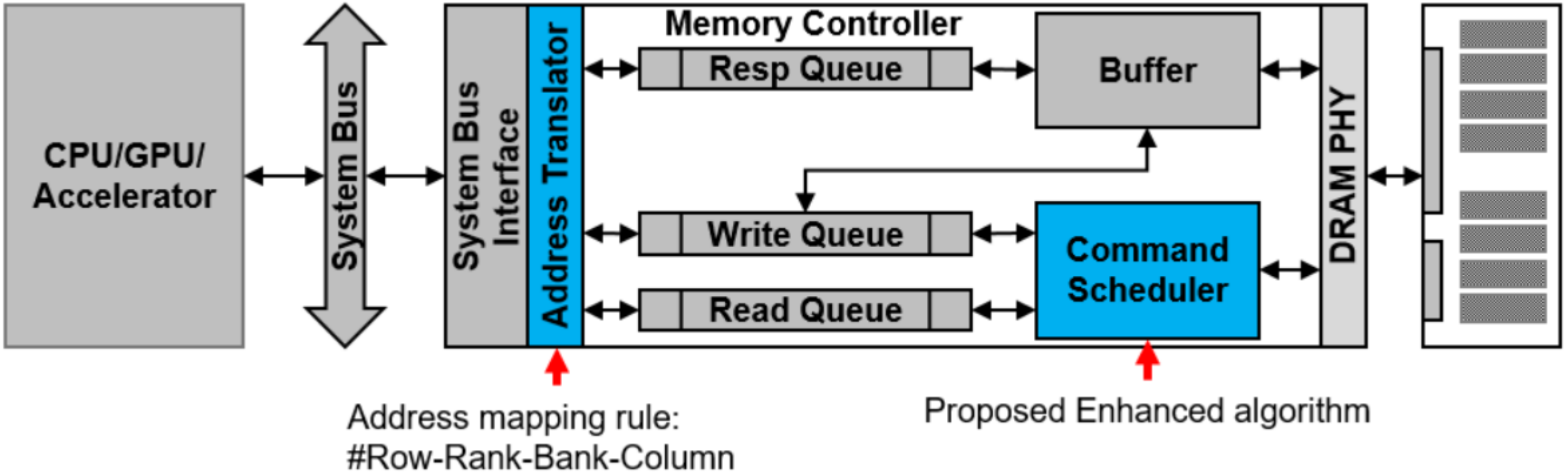
DRAMA – Command Scheduler



- Define 2 time thresholds
  - $t_{min\_threshold} = t_{RFCpb}$ (Time refresh per-bank)
  - $t_{max\_threshold} = t_{REFI}/8$ (Refresh Interval / number of bank)
- Strategy
  - Lock a refreshing bank until $t_{min\_threshold}$ is expired
  - Access non-locked banks during $t_{max\_threshold}$ → Refresh overhead is hidden

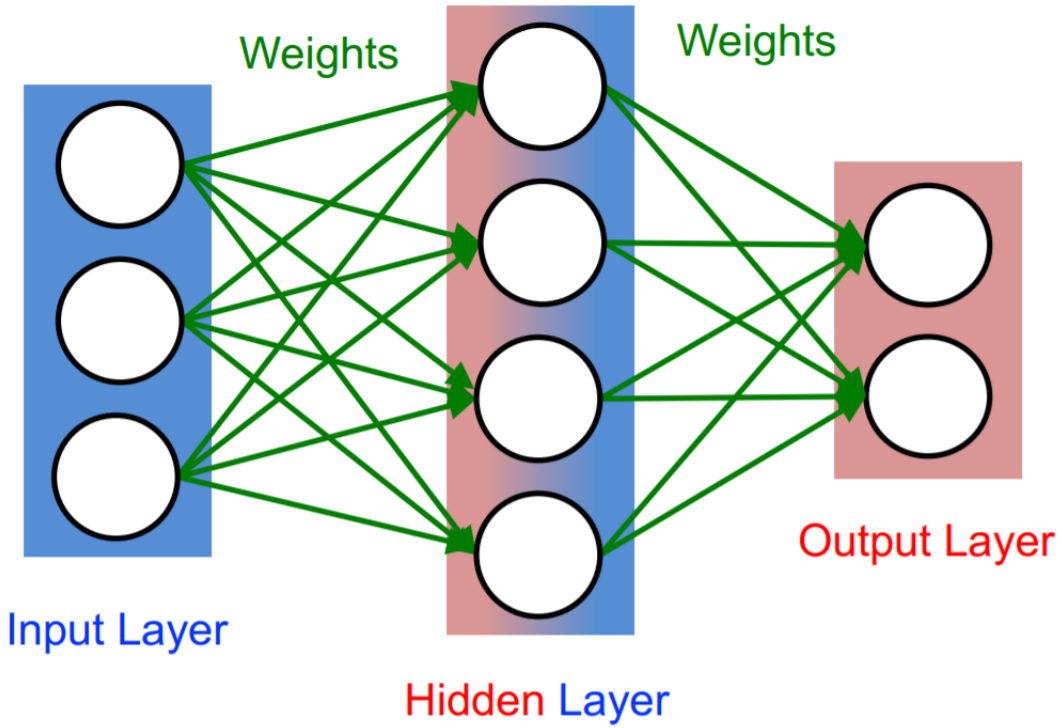
Bank/Per-Bank Refresh overhead
Bank RD/WR overhead

DRAMA Memory Controller

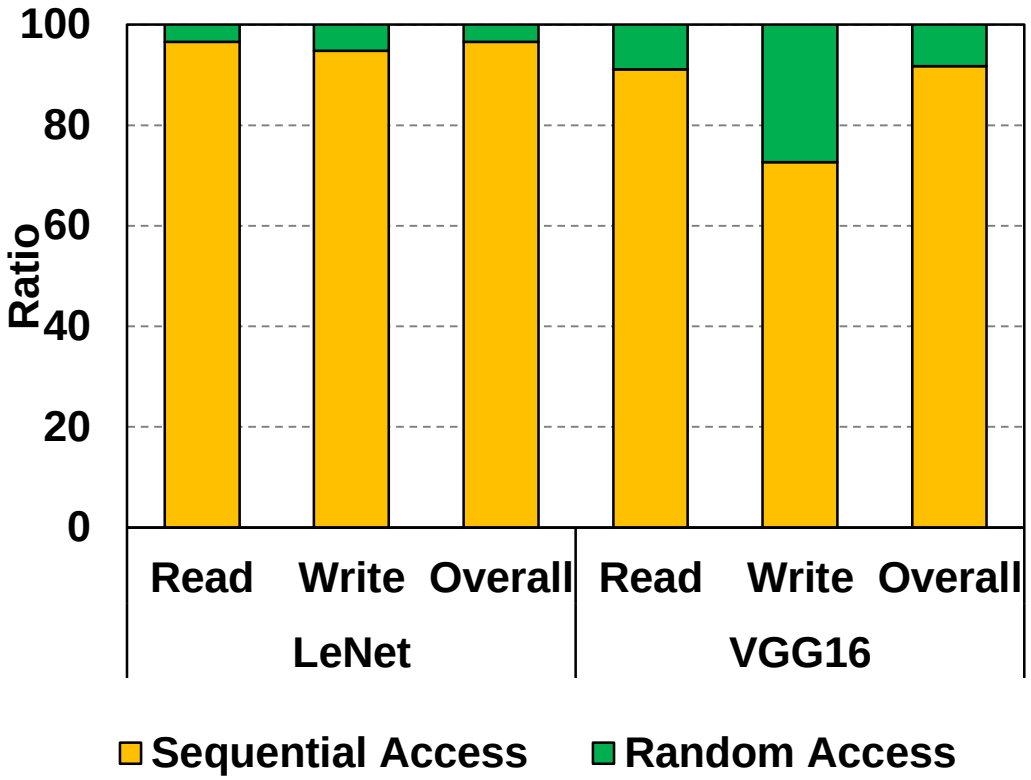


- **Two major changes in the DRAM Controller to support the per-bank refresh**
  - **Command Scheduler**
  - **Address Translator**

Fetching DNN's Data from DRAM

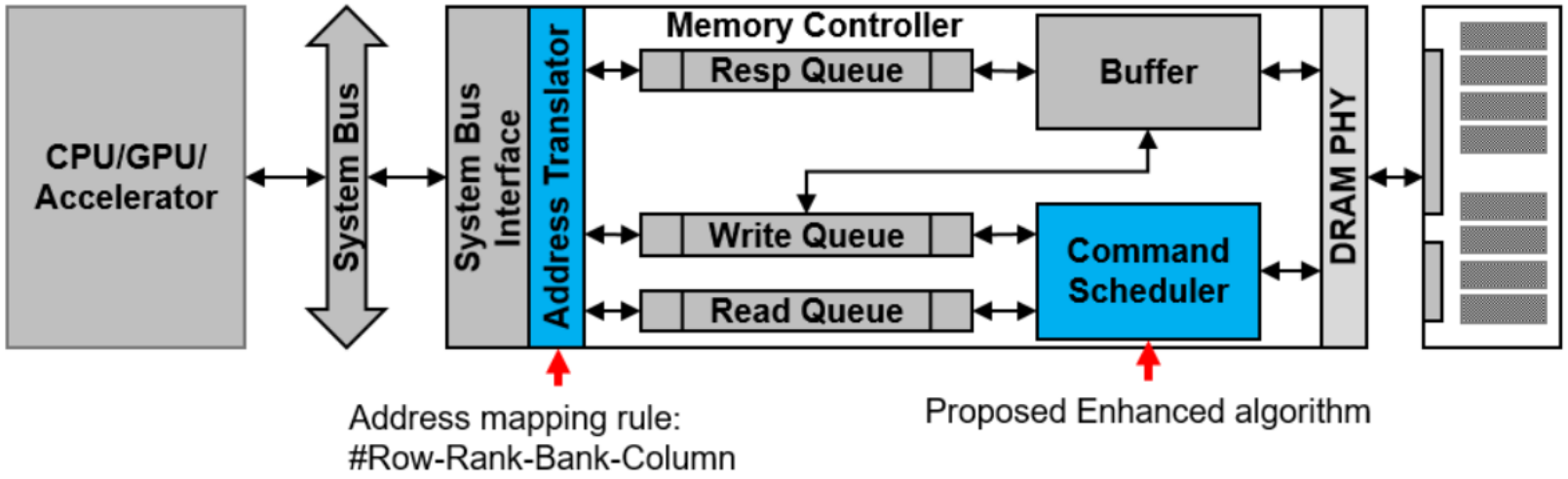


Vivienne Sze et al. 2019



- Layer-by-layer computation
  - The output data of a certain layer become the input data of the following layer
  - Sequential DRAM Accesses are dominant

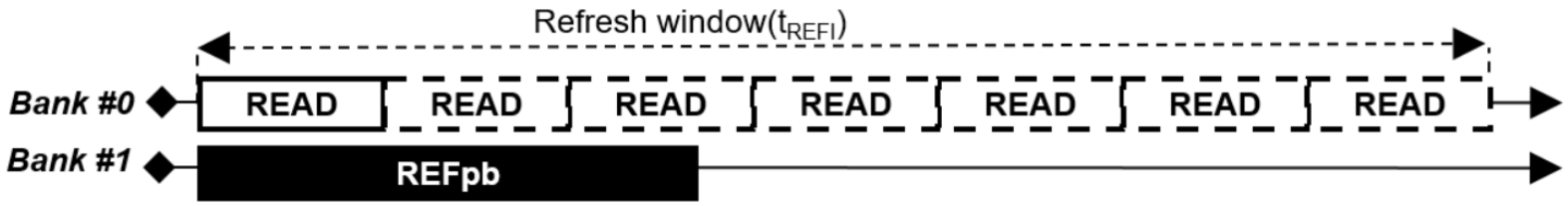
Our Address Translator



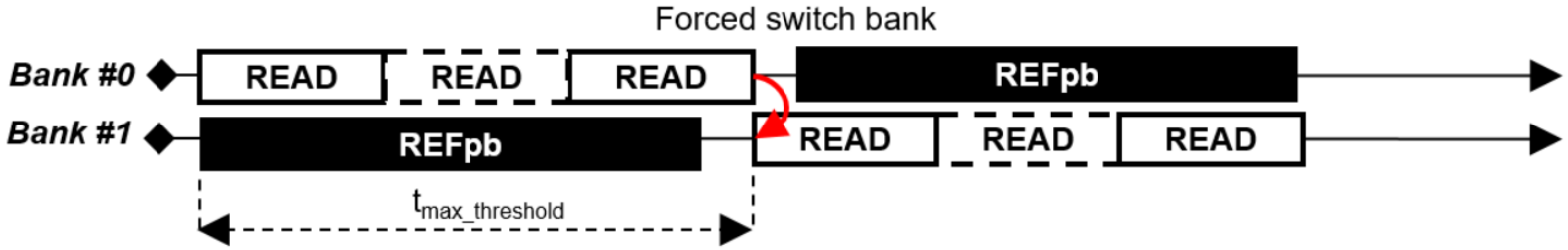
- 2 Major changes in DRAM Controller
  - –Command Scheduler
  - –Address Translator

- Scheduling to consider the sequential data access
  - Conventional:
    - #Row-Rank-Bank-Column
  - Our proposed scheme:
    - #Row-Rank-Column[9:7]-Bank-Column[6:0]

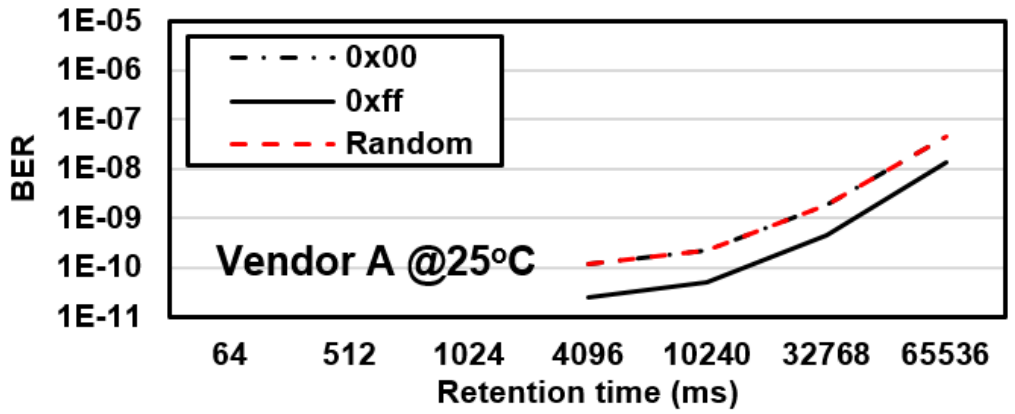
a. Conventional address translator

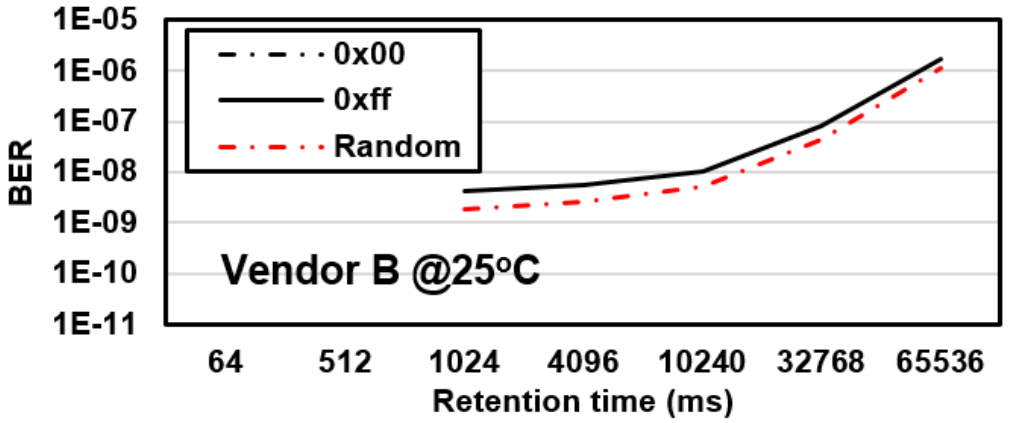


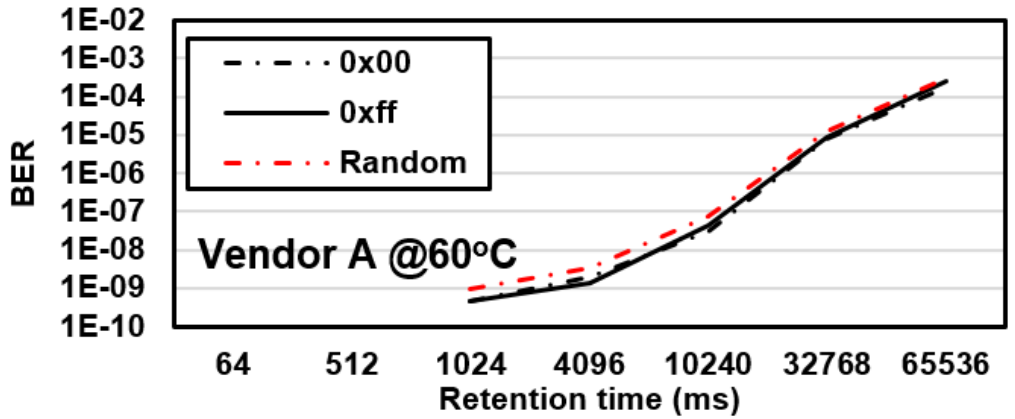
b. Our proposed address translator

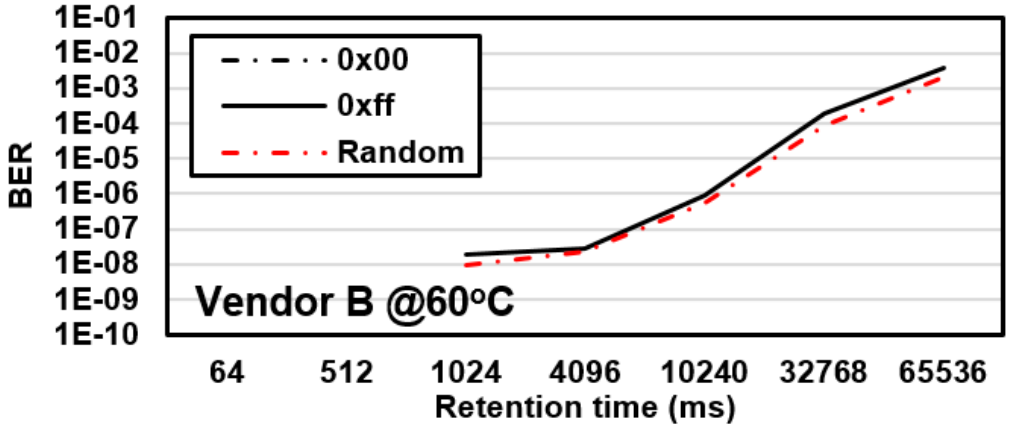


Validation - Setup





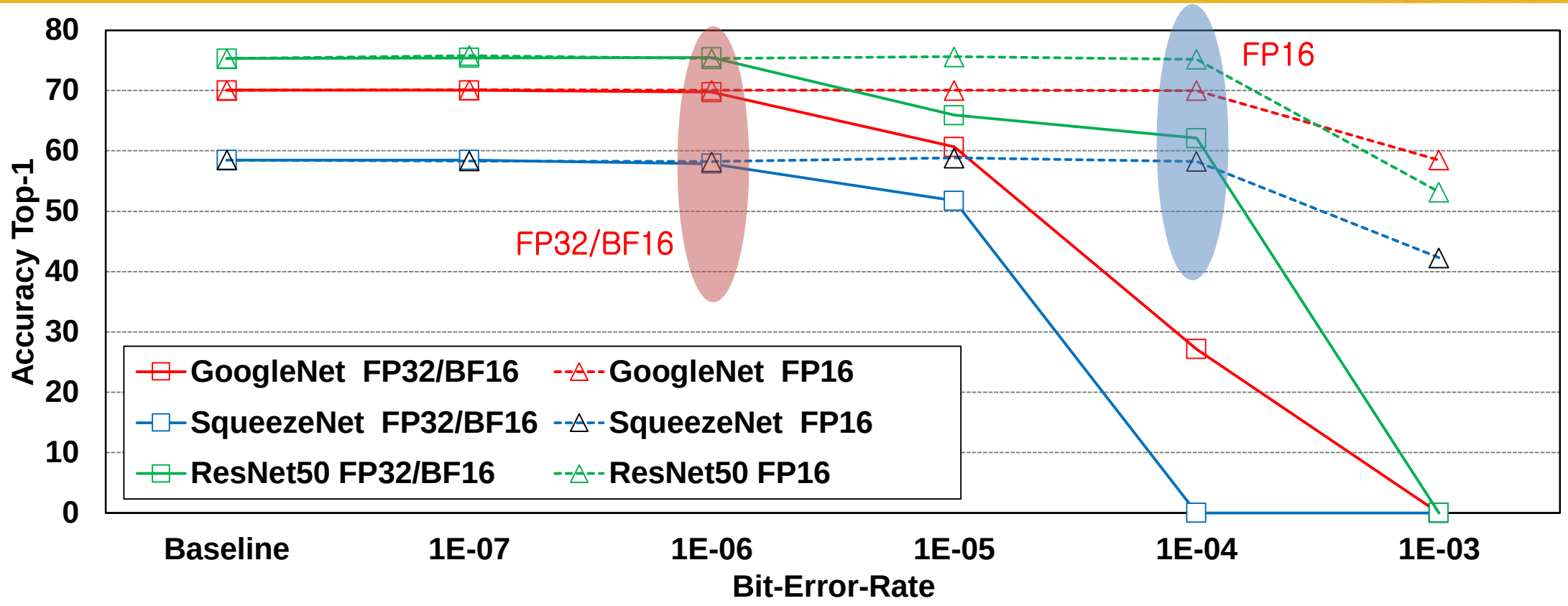




- The working temperature is less than 60°C in the data-center (**)
  - Inject 10^{-3}~10^{-7} BER to weights, activations, gradients and biasings during forward/backward phase
- We find the safe-BER threshold to hardly affect the training accuracy
  - Extract the maximum refreshing time during training by using the extracted empirical BER model from DDR4

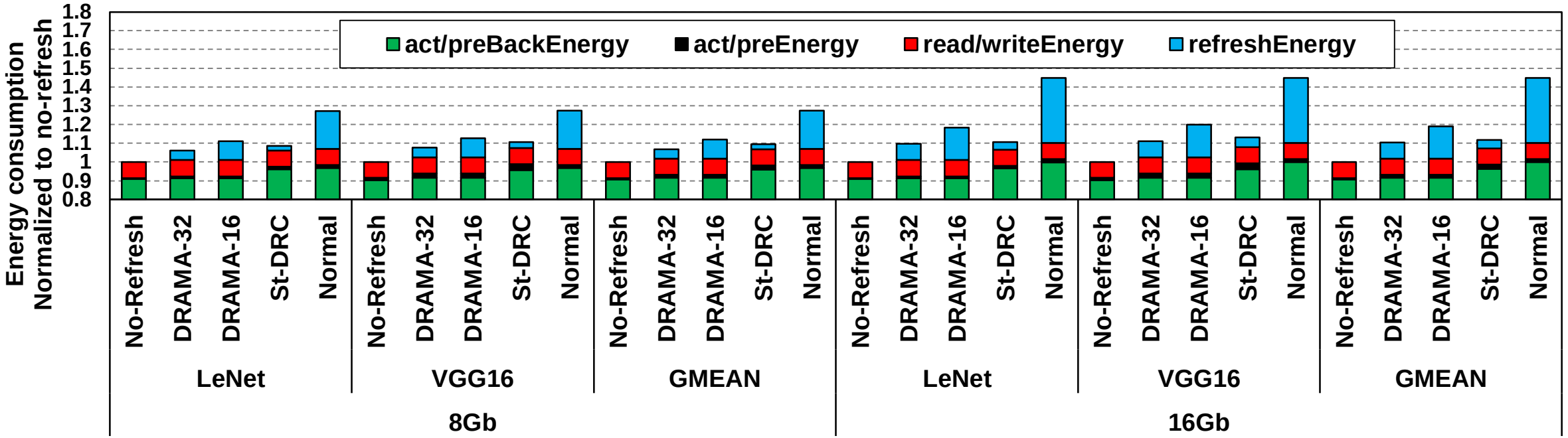
(**)Donghyuk Lee, Adaptive-latency DRAM: Optimizing DRAM timing for the common-case. HPCA 2015

Validation – Training process

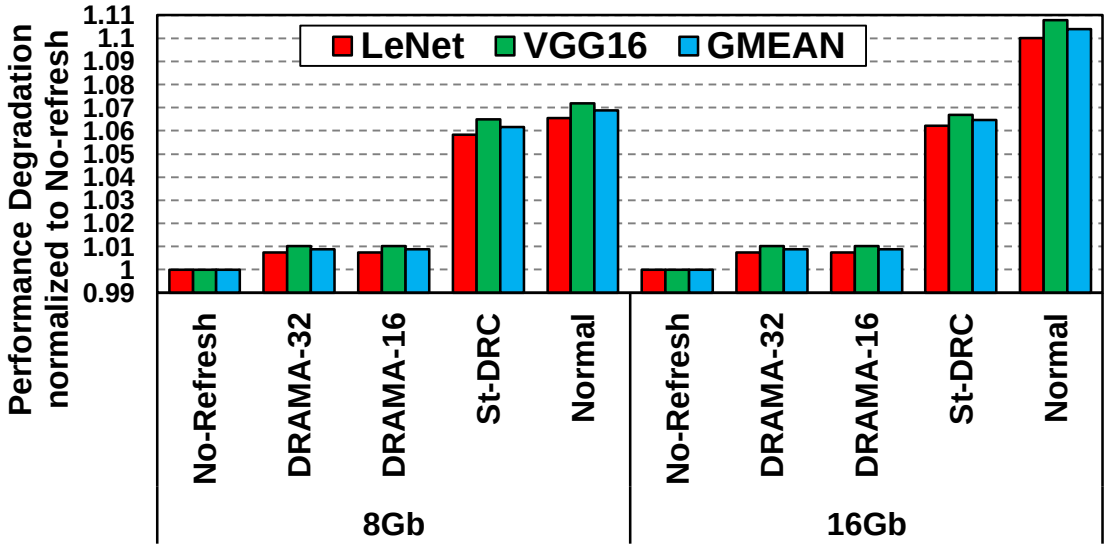


- Safe BER-training threshold
  - – 10^{-6} FP32/BF16
  - – 10^{-4} FP16
- → Maximum refreshing time during training:
  - – 10sec @ 60ºC

Energy and Performance Simulation Results



- LeNet+VGG16 simulation on GEM5 with different size of DRAM
  - DRAM energy reduces 16%
  - Performance improves 10.4%



Summary

- Our DRAMA can:
  - Improve both performance and energy reduction for Deep Training
    - Deliver a near optimal performance improvement and energy reduction in DNN training
  - Provide system level approaches with minimal modifications in the physical DRAM chip
    - Fully comply with the JEDEC standard
  - Address translator and command scheduler ensure the command scheduling constraints are met while enabling per-bank refresh.

Acknowledgment & Thank you

- This research was supported by
  - –National R&D Program through the National Research Foundation of Korea(NRF) funded by Ministry of Science and ICT (2020M3F3A2A01085755).
  - –Ministry of Trade, Industry and Energy through the Korea Semiconductor Research Consortium support program for the development of the future semiconductor device under Grant 10080594.

- Any question?