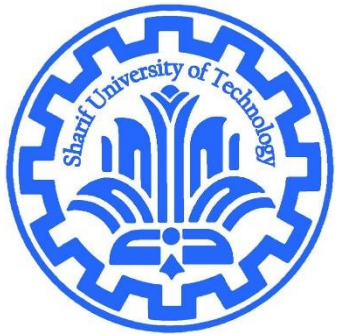
MLC STT-RAM for Deep Neural Networks Accelerators

Masoomeh Jasemi

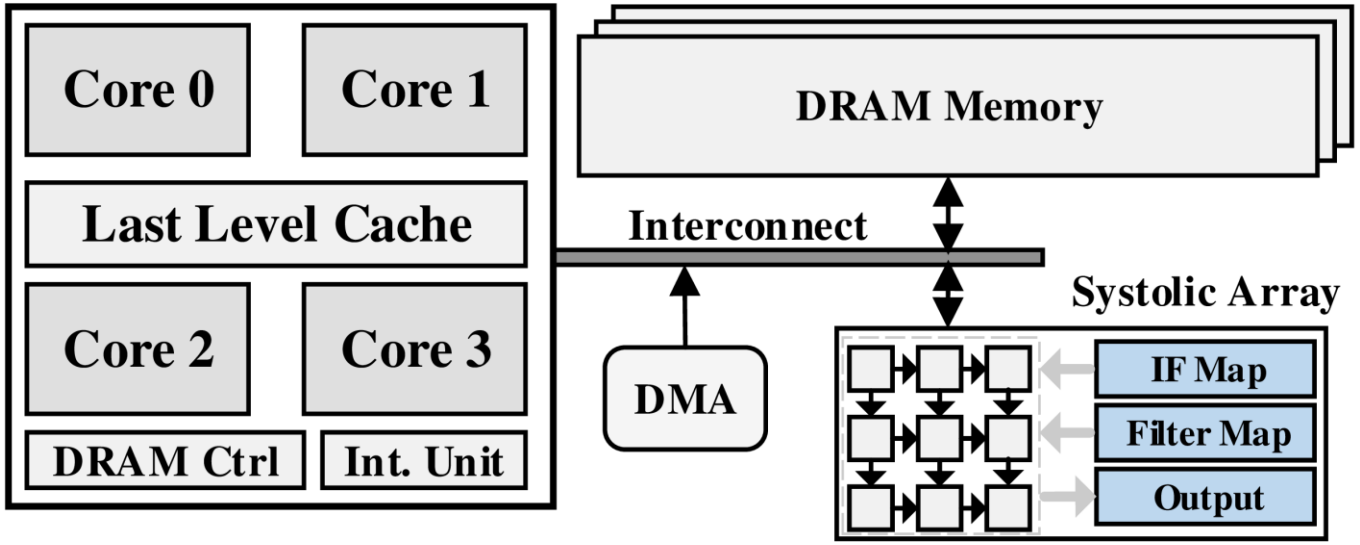
mjasemi@uci.com

jasemi@ce.sharif.edu



Accelerator based architecture for CNNs

- Deep Neural Networks for energy constrained tasks
  - General purpose processors cannot perform CNN computation
  - Accelerators
    - Improve performance of CNNs architecture
    - Systolic array is a good candidate
      - Fast matrix to matrix operation
      - Fast matrix to vector operation
      - Larger on-chip memory is required!



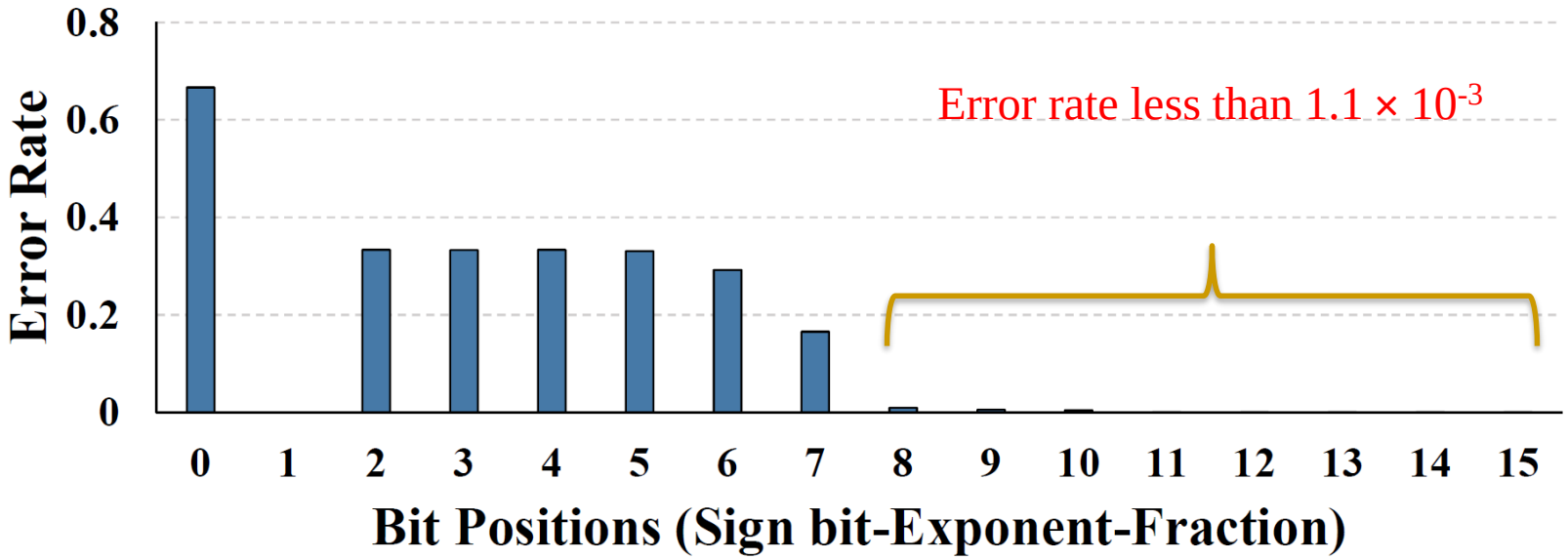
MLC STT-RAM

- Pros:
  - High density
  - Low leakage power
  - Close to SRAM read latency
- Cons:
  - Higher error rate: Specially for MLC mode
  - Longer write operation

Device	SRAM	STT-RAM
Density	1X	4X
Read time	Very fast	Fast
Write Time	Very fast	Slow
Read power	Low	Low
Write power	Low	High
Leakage power	High	Low
Endurance	10^{16}	4×10^{12}

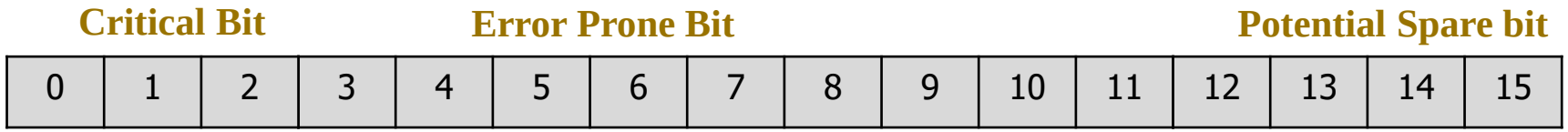
Impact of bit flips on 16-bit floating-point

- IEEE-754 standard half-precision floating-point format
  - Very sensitive to single bit flip error
- Monte-Carlo experiments
  - 100 million 16-bit FP numbers in the range of (-1,1)
  - Injecting error based on uniform random distribution
  - Calculate Sum of Squared Errors (SSE)



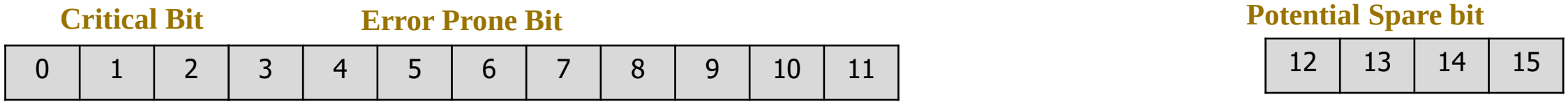
Proposed Scheme

- Creating a backup for critical bits
- Dividing binary representation of each 16-bit FP:
  - Critical Bit
  - Error Prone Bit
  - Potential Spare Bit
  - Copy MSBs of Critical Bits to Potential Spare Bit
  - Change Critical Bits and Potential Spare Bit regions to SLC



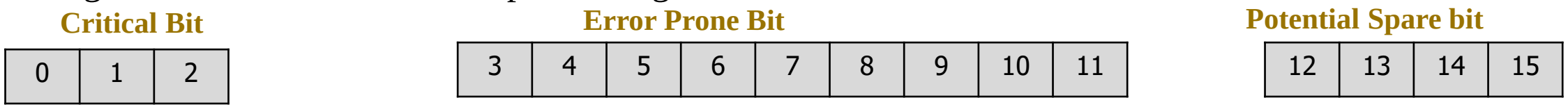
Proposed Scheme

- Creating a backup for critical bits
- Dividing binary representation of each 16-bit FP:
  - Critical Bit
  - Error Prone Bit
  - Potential Spare Bit
  - Copy MSBs of Critical Bits to Potential Spare Bit
  - Change Critical Bits and Potential Spare Bit regions to SLC



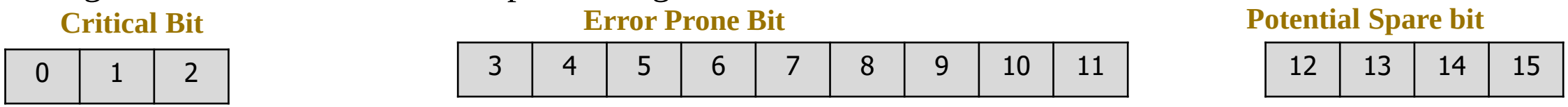
Proposed Scheme

- Creating a backup for critical bits
- Dividing binary representation of each 16-bit FP:
  - Critical Bit
  - Error Prone Bit
  - Potential Spare Bit
  - Copy MSBs of Critical Bits to Potential Spare Bit
  - Change Critical Bits and Potential Spare Bit regions to SLC



Proposed Scheme

- Creating a backup for critical bits
- Dividing binary representation of each 16-bit FP:
  - Critical Bit
  - Error Prone Bit
  - Potential Spare Bit
  - Copy MSBs of Critical Bits to Potential Spare Bit
  - Change Critical Bits and Potential Spare Bit regions to SLC



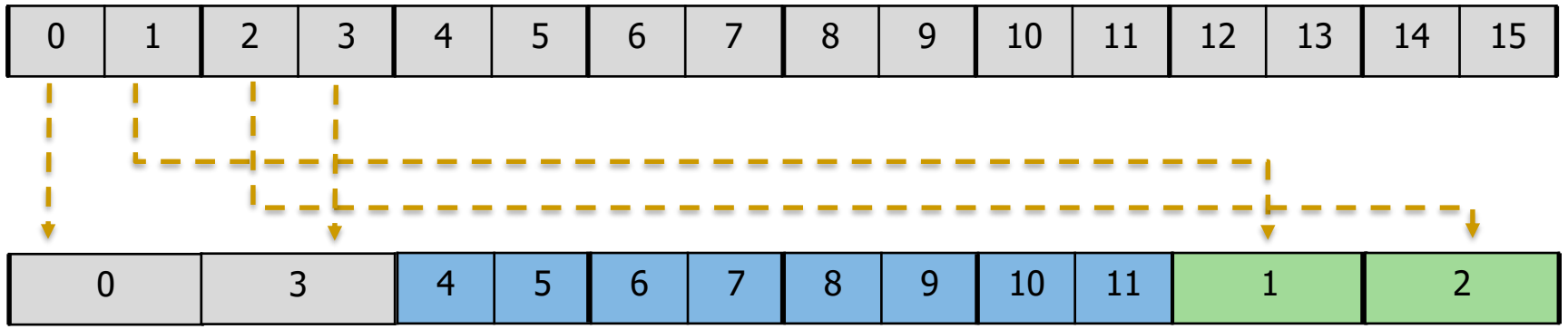
How large critical bit (CB) region should be?

Read & Write

- Write
- Read

Read & Write

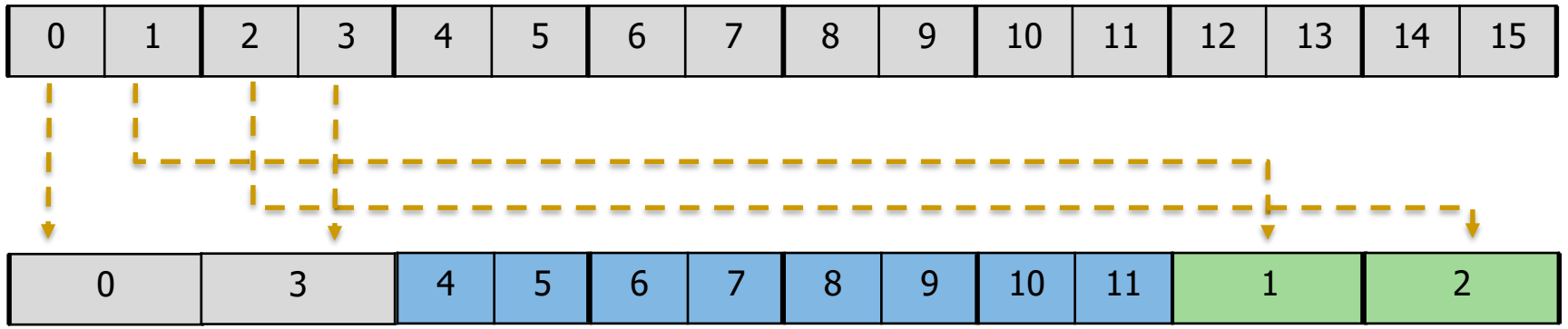
- Write



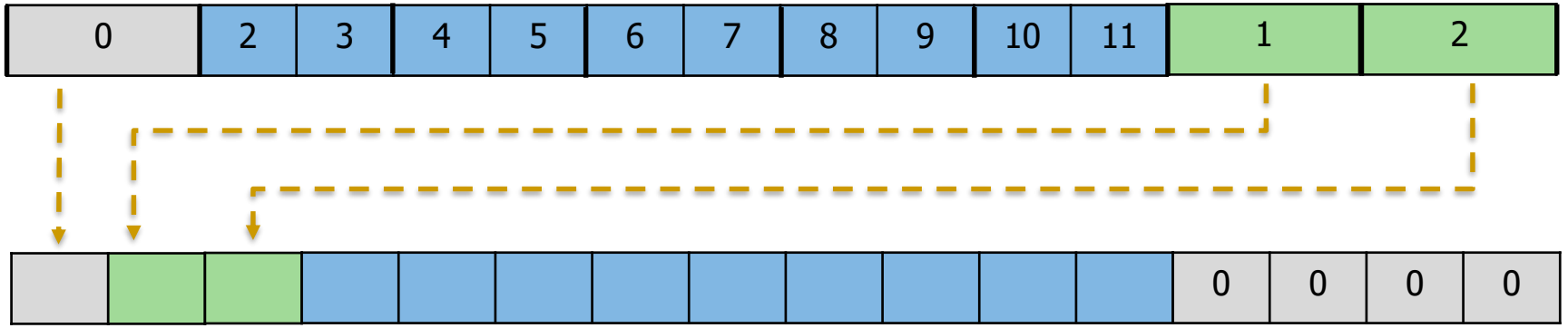
- Read

Read & Write

- Write

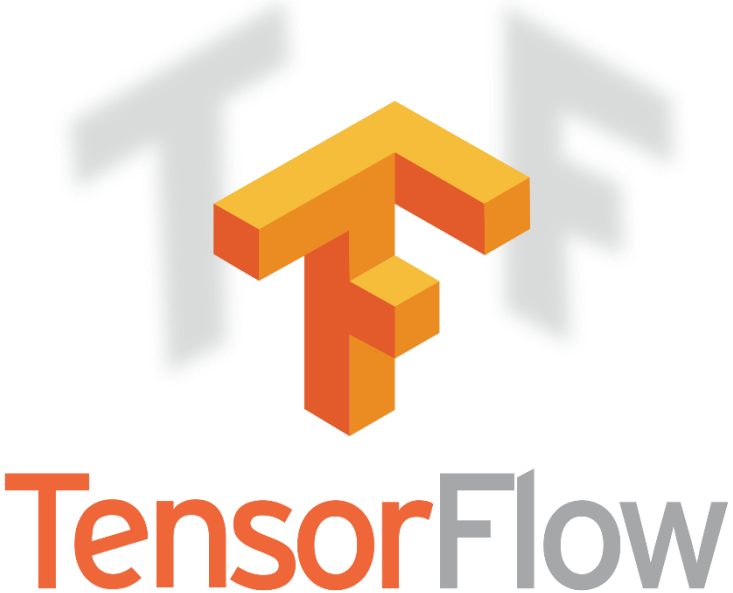


- Read



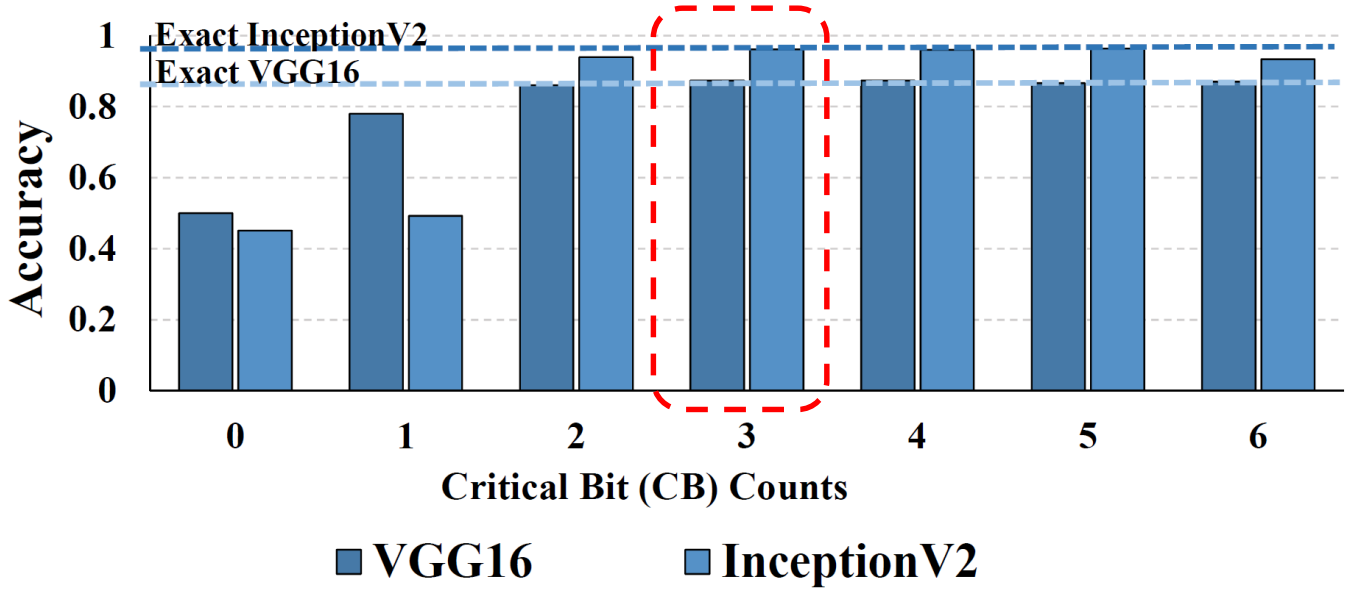
Methodology

- Methodology:
  - Google Tensorflow: Run CNN architectures with faulty weights
  - Training with exact numbers
  - Inference with injected faults in weights
- Workloads:
  - VGG-16 (Imagenet)
  - Inception V2 (Imagenet)



Evaluation results

- Metric
  - Accuracy of prediction
    - Baseline
    - Error free memory subsystem
    - Pure MLC STT-RAM
    - Best choice: CB = 3



Other repacking scheme

- Shift
- Rotating
- Clustering

Any Question?