AI Inference Memory System Tradeoffs

When companies describe their AI Inference chip they typically give TOPS but don't talk about their memory system which is equally important.

What is TOPS? It means Trillions or Tera Operations per Second. It is primarily a measure of the maximum achievable throughput but not a measure of actual throughput. Most operations are MACs (multiply/accumulates), so TOPS = (number of MAC units) x (frequency of MAC operations) x 2.

To actually make good use of the TOPS the chip needs a memory system that can keep the MACs busy most of the time (high MAC utilization): this is the key to actually achieving high throughput.

The memory system for an AI Inference chip needs to

1. Provide capacity for storage of the Weights for the neural network model; for the storage of the code for executing the neural network model; and for storage of the initial input/image and intermediate activations
2. Provide weights and intermediate activations to the MACs at a bandwidth high enough to keep up with the MAC execution rate
3. Write back intermediate activation outputs back to memory at a bandwidth sufficient not to stall operation waiting on completion of a write operation

How much memory capacity is required?

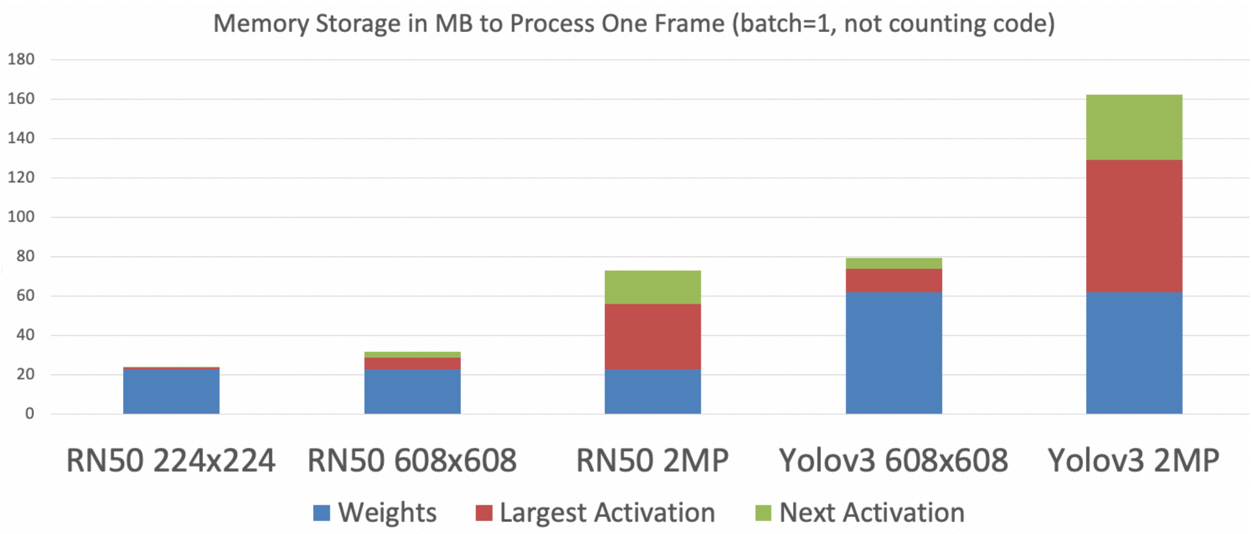
Weights take a lot of room: 22.7MB for ResNet-50 INT8 and 62MB for YOLOv3 INT8. The expectation is that future, better models will be bigger models with more weights.

Intermediate activation storage: the largest activation for ResNet-50 is 0.8MB and the next largest is 0.4MB; this is for 224x224 images. So a buffer memory of 1.5-2MB is sufficient for batch size =1. If batch size = N, the buffer size needs to be N times larger than 1.5-2MB. So batch size = 16 means the buffer size is 24-32MB making it larger than the capacity required to store weights.

Intermediate activation storage for YOLOv3 can be much larger depending on the size of the image. YOLOv3 requires intermediate activation storage of ~18MB for 608x608 images and >100MB for 2Megapixel images. Because these buffer sizes are so large, doing batching does not make sense.

Code size is significant but no one is disclosing this information yet for their chips.

The chart below shows the total Megabytes required in an inference chip for weights and activations for ResNet-50 and YOLOv3 at various images sizes.



There are 3 choices for memory system implementation for AI Inference chips. Most chips will have a combination of 2 or 3 of these choices in different ratios:

1. Distributed local SRAM – a little less area efficient since overhead is shared across fewer bits, but keeping SRAM close to compute cuts latency, cuts power and increases bandwidth.
2. Single bulk SRAM – a little more area efficient but moving data across chip increases power, increases latency and makes the single SRAM the performance bottleneck.
3. DRAM – much cheaper cost per bit but the number of bits is likely way more than is needed; the power is significantly higher than SRAM access; and the cost on the controller to access the DRAM with high bandwidth is very significant.

Let’s look at some specific numbers.

In 16nm, 1MB of SRAM is about 1.1mm2. Using <30MB it’s possible to have a memory system (excluding code size) that can hold the weights and activations, for batch=1, for ResNet-50 on-chip without DRAM. So a non-DRAM chip is possible for simple models and small image sizes.

But image sensors come in 1, 2 and 4 Megapixels and larger images result in higher prediction accuracy for any model.

The benchmark model most commonly requested is YOLOv3 with 2Megapixel images: this would require ~160MB of SRAM or ~180mm2 in 16nm to keep the weights and activations on chip (this is before code storage is factored in).

To make AI Inference cost-effective at the edge, it is not practical to have almost 200mm2 of SRAM. Instead, some of the memory capacity needs to be DRAM.

DRAM is lowest cost-per-bit but the smallest DRAM available today for LPDDR4 is 4Gigabits = 512MB: much more capacity than needed for a single model above. (The extra capacity can be used to store weights/code for multiple models so the inference accelerator can switch between models rapidly).

To be useful in keeping MACs “fed”, DRAM bandwidth needs to be high. DRAMs need to be x32 databus per device at 3 or 4 Gigatransfers/second.

This is not cheap to connect to. The silicon area for x 32 LPDDR4 PHY plus memory controller is 5-6mm2 in 16nm; and the BGA balls required is 100-150 balls. Some AI Inference chips have 256-bit buses which means 40-48mm2 for PHY/controller and 800-1200 balls! The package cost for big BGAs can rival the silicon cost.

The ideal AI Inference accelerator will be one that can achieve high MAC utilization with 1 or 2 DRAMs and the least on-chip SRAM in order to keep costs and power lower while achieving high throughput.

As you evaluate AI Inference options, ask about the size of on-chip SRAM and the bus width for off-chip DRAM and keep in mind the cost incurred to achieve the capacity and bandwidth. This is just as or is more important than knowing the TOPS the chip has.

Geoff Tate
CEO Flex Logix