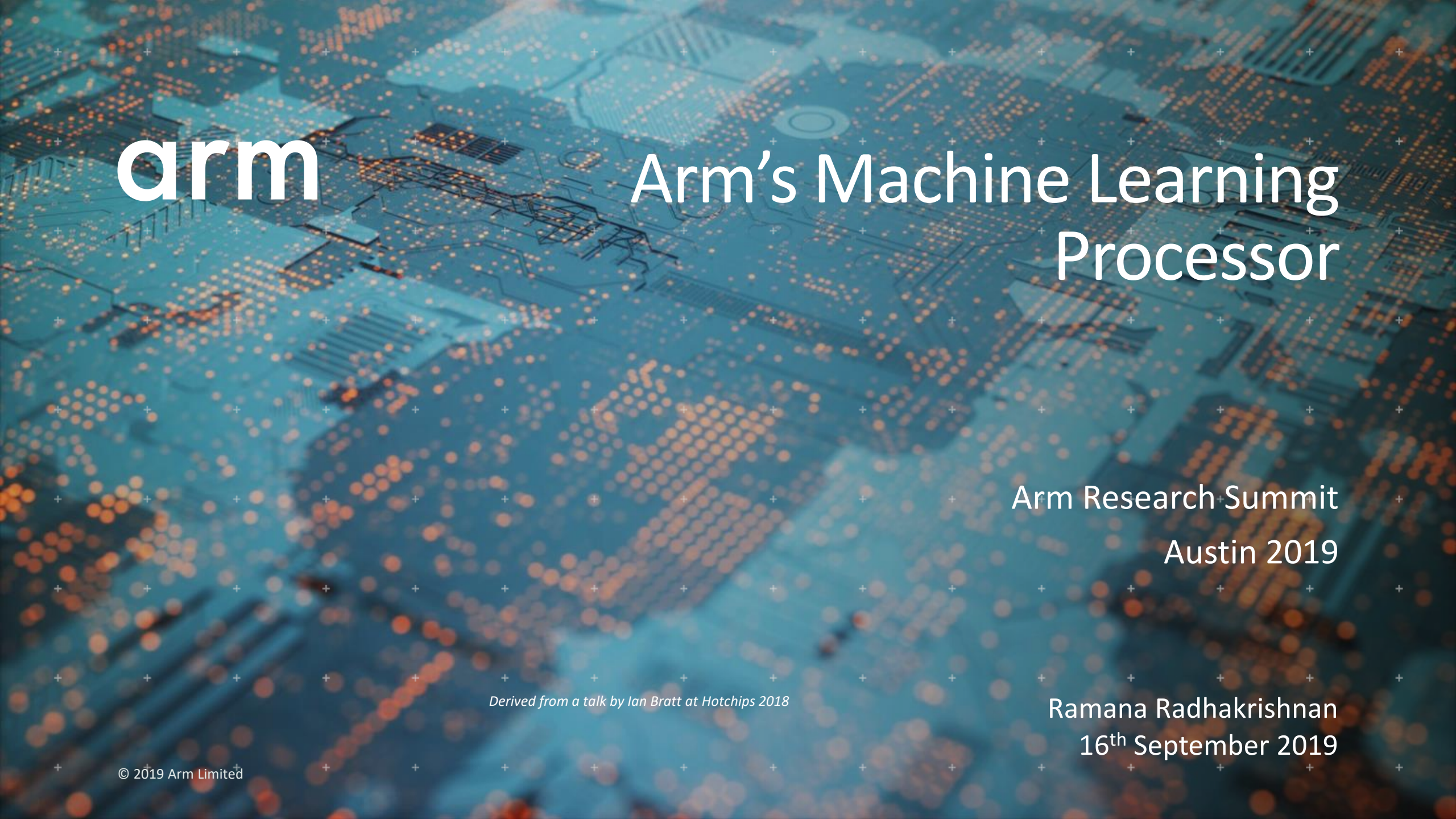


Arm's Machine Learning Processor

Arm Research Summit

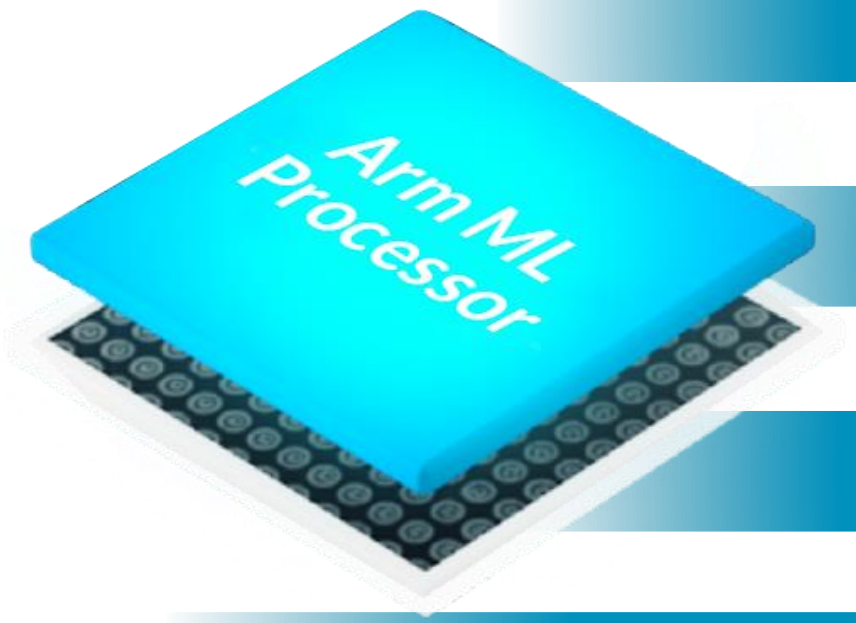
Austin 2019

Derived from a talk by Ian Bratt at Hotchips 2018

Ramana Radhakrishnan

16th September 2019

Introducing the Arm Machine Learning (ML) Processor



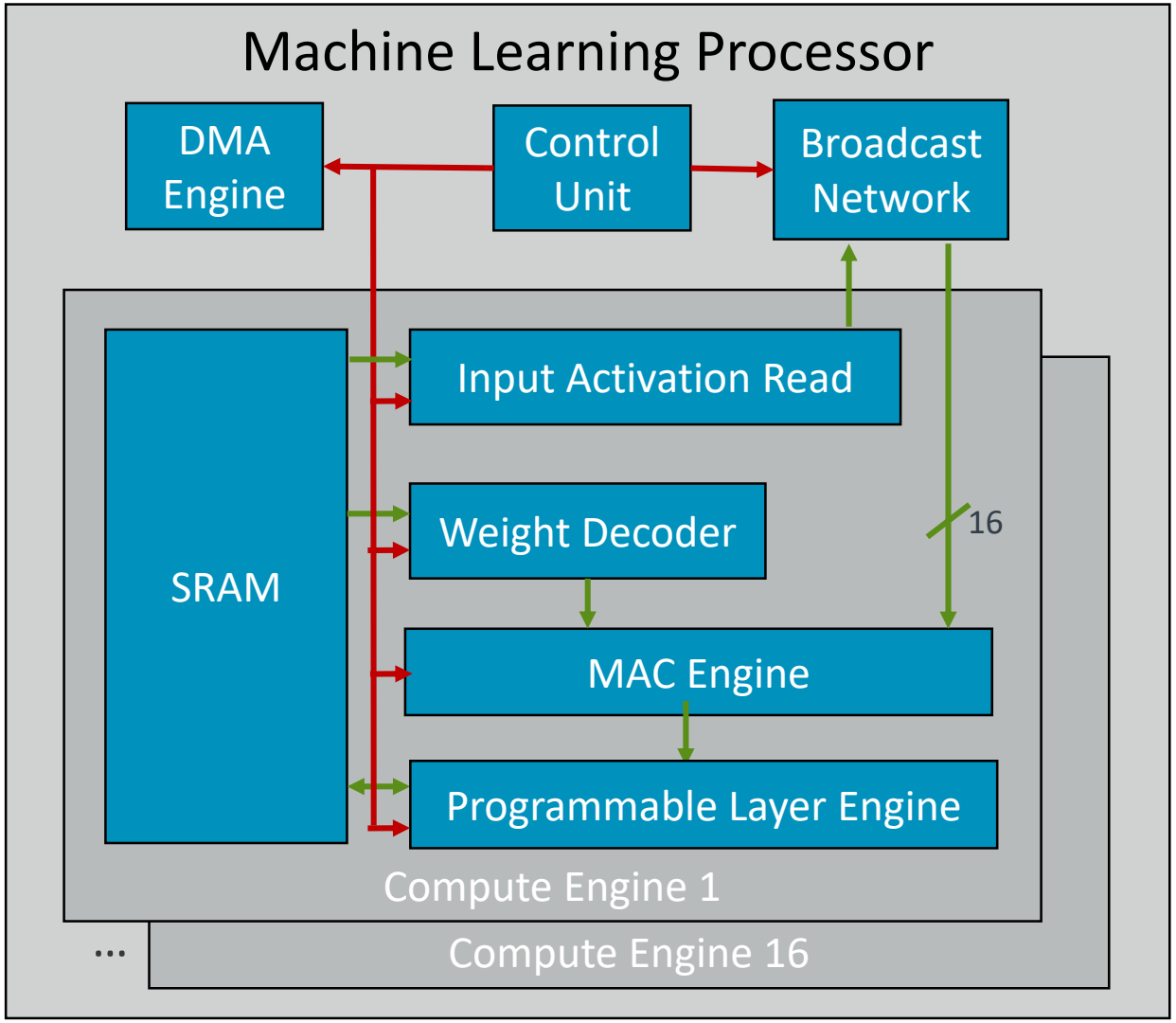
- Optimized ground-up architecture for machine learning processing
- Massive efficiency uplift from CPUs, GPUs and DSPs
- Open-source stack enables easy deployment
- Architecture scales from IoT to server and automotive
- First design targets mobile with derivatives for additional segments

2

arm

Arm's ML Processor

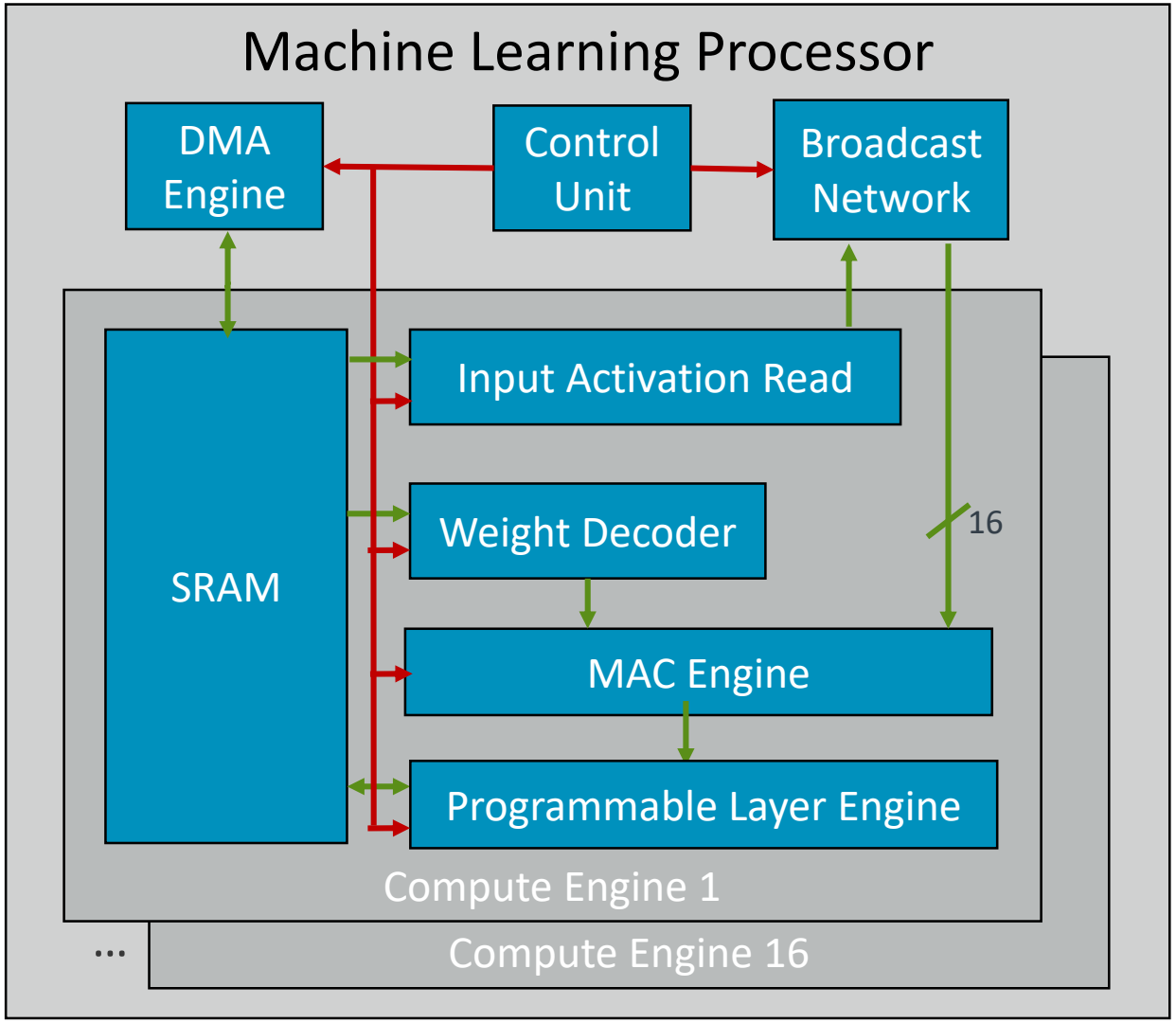
- 16 Compute Engines
- ~4 TOPS of convolution at 1 GHz.
- 8 bit quantized integer support
- 1MB of SRAM
- Support for Android NN API and Arm NN
- Optimized for CNNs, RNN support



arm

4 Key Ingredients for a Machine Learning Processor

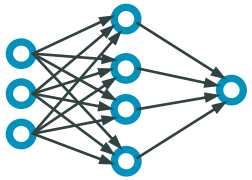
- Static scheduling
- Efficient convolutions
- Bandwidth reduction mechanisms
- Programmability/flexibility



arm

Arm's ML processor: Static Scheduling

- CNNs are statically analyzable
- Compiler takes a NN and maps it to a command stream consumed by the ML processor



NN Compiler

Command Stream

DMA X
DMA Y
WAIT for DMA (X,Y)
Conv X, Y
etc
...

ML Processor

DMA Engine

Control Unit

SRAM

Weight Decoder

MAC Engine

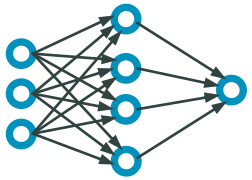
Programmable Layer Engine

Compute Engine 1

Compute Engine 16

Arm's ML processor: Static Scheduling

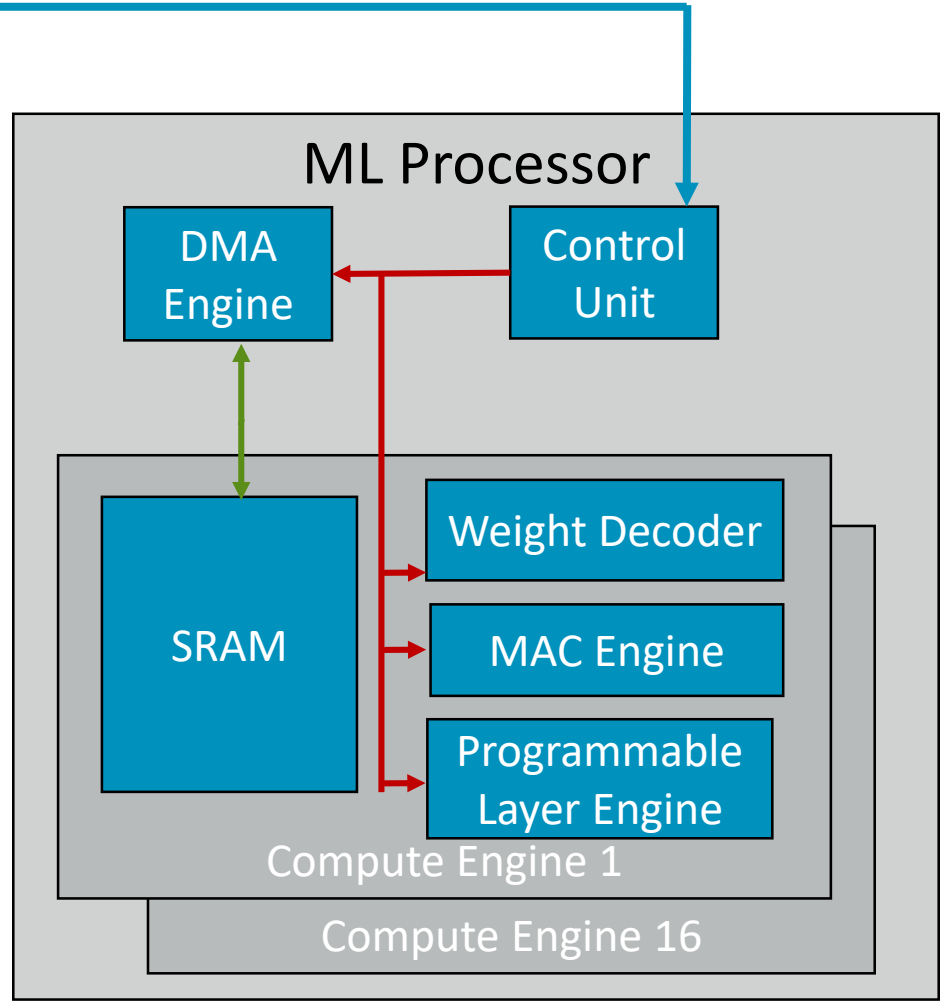
- No caches
- Simplified flow control
- Simplified hardware (but requires careful co-design with the compiler)
- Relatively predictable performance



NN Compiler

Command Stream

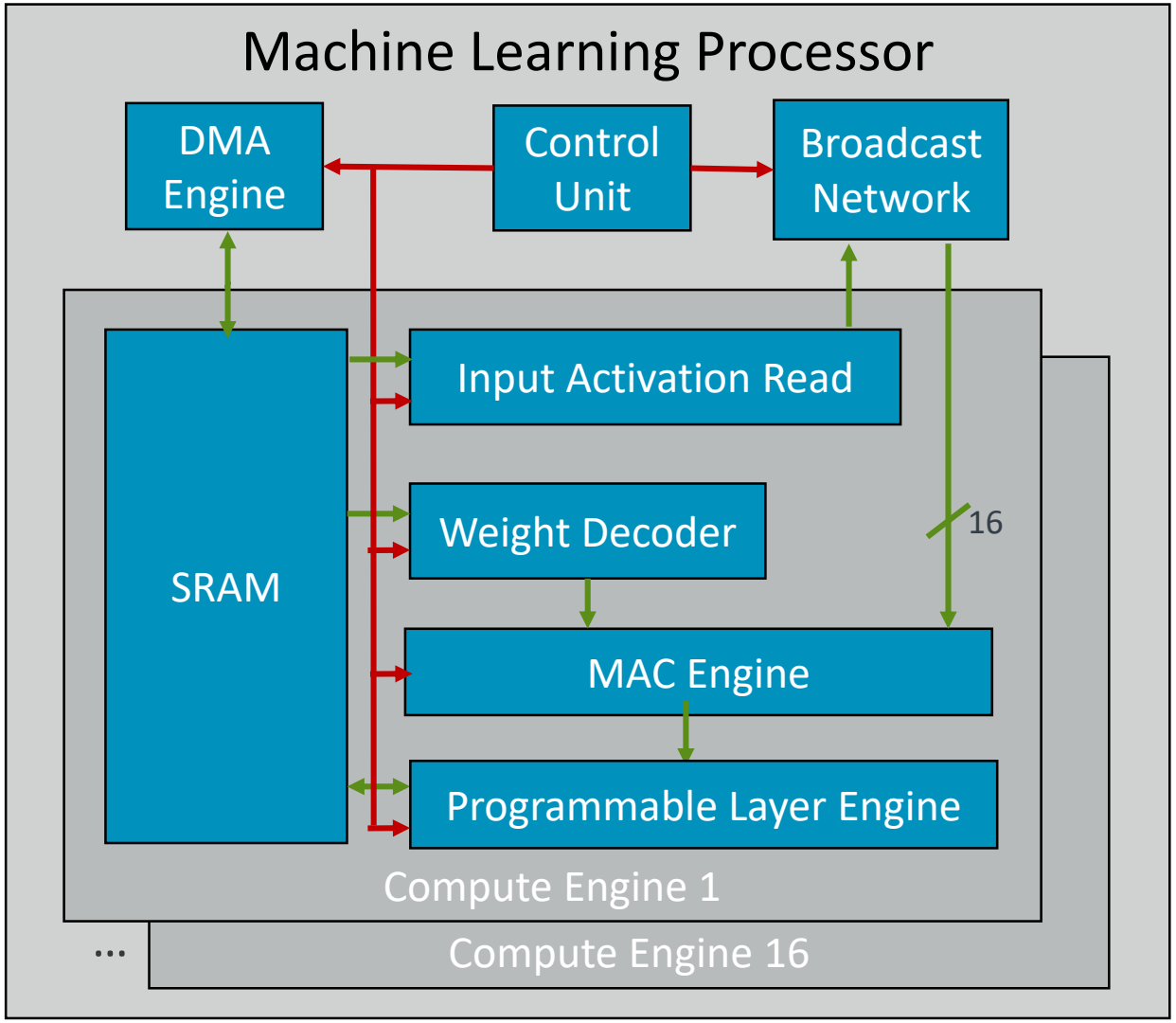
DMA X
DMA Y
WAIT for DMA (X,Y)
Conv X, Y
etc
...



arm

4 Key ingredients for a Machine Learning Processor

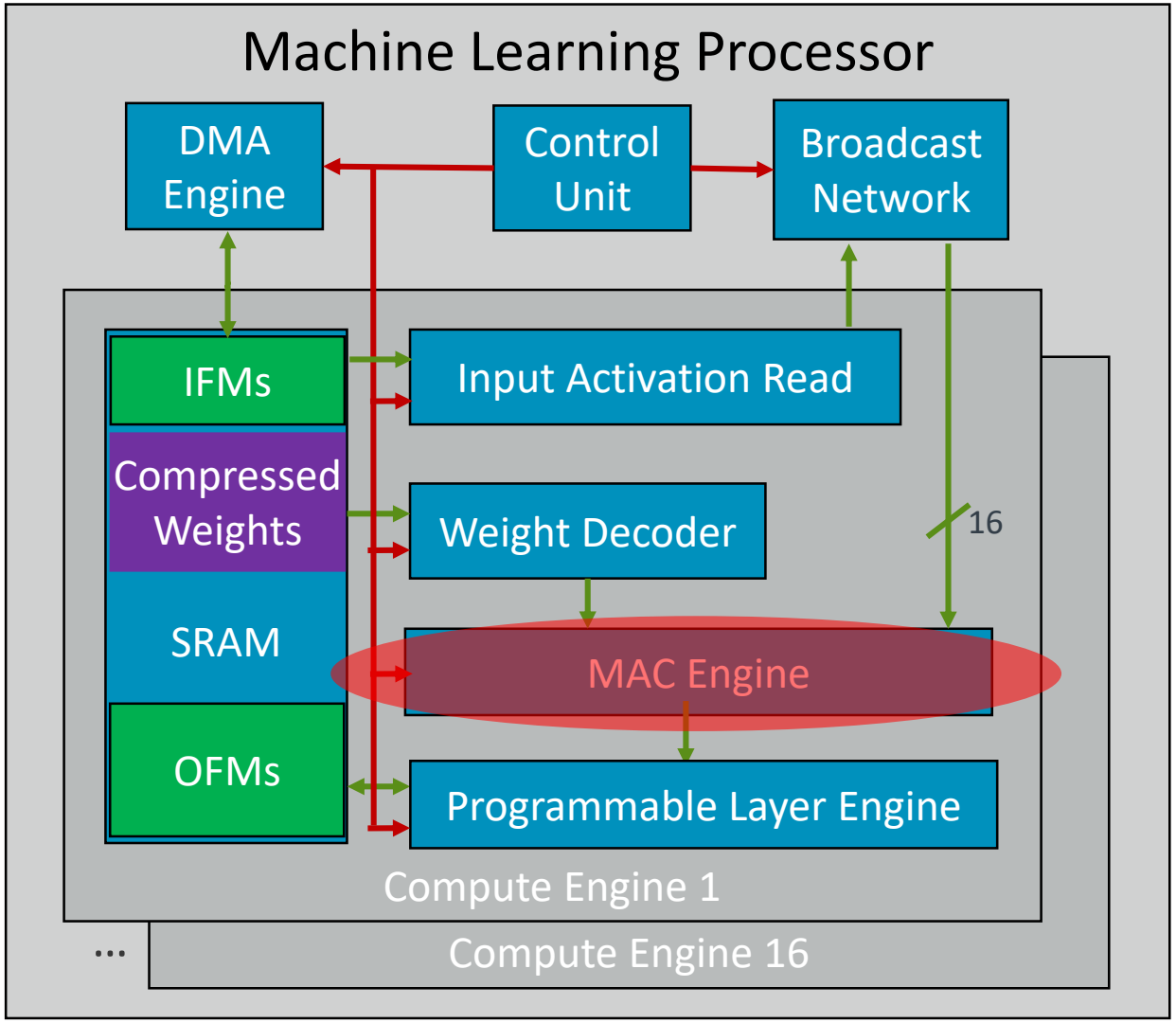
- Static scheduling
- Efficient convolutions
- Bandwidth reduction mechanisms
- Programmability/flexibility



arm

Convolutions

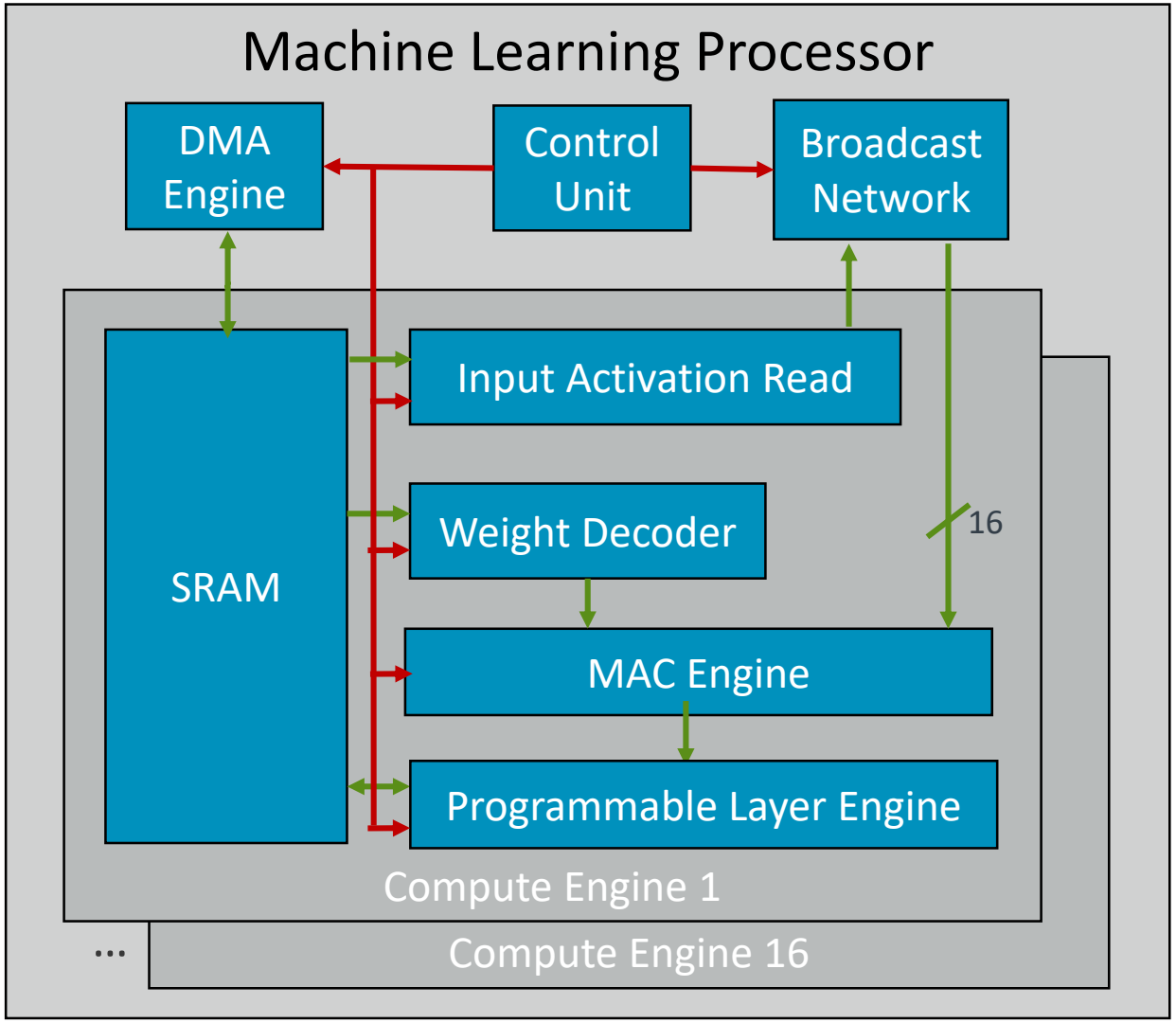
- MAC Engine capable of eight 16x16 8-bit dot products
  - MAC Engine = 2 * 8 * 16 = 256 ops/cycle
  - 16 MAC Engines. = 16 * 256 = 4096 ops/cycle
  - 4.1 TOPs @ 1 GHz
  - 32b accumulators
- The utilization of the MAC engine depends on conv parameters
- Datapath gating for zeros (~50% power reduction



arm

4 Key ingredients for a Machine Learning Processor

- Static scheduling
- Efficient convolutions
- Bandwidth reduction mechanisms
- Programmability/flexibility

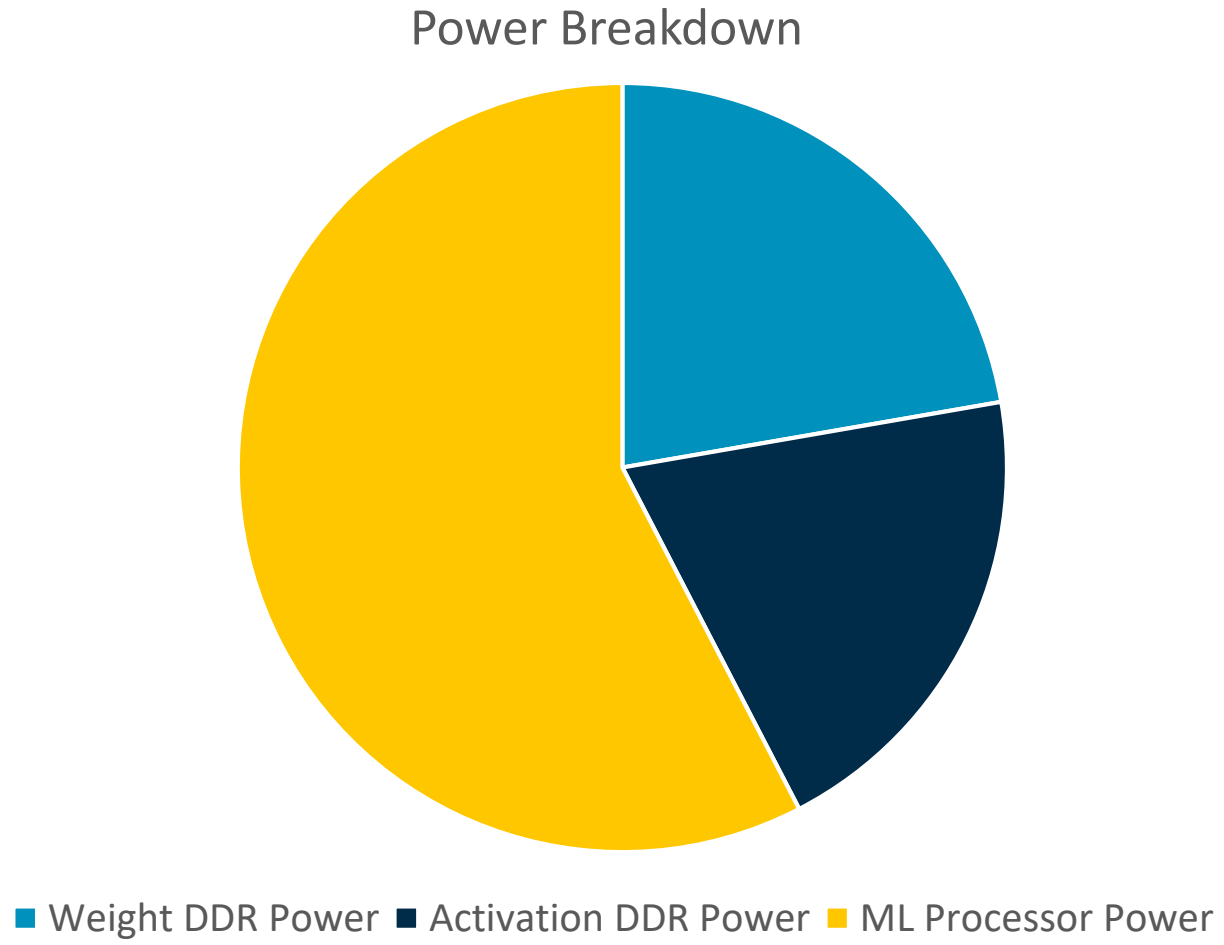


arm

Importance of Weight and Feature Map Compression

- DRAM power can be nearly as high as the processor power itself

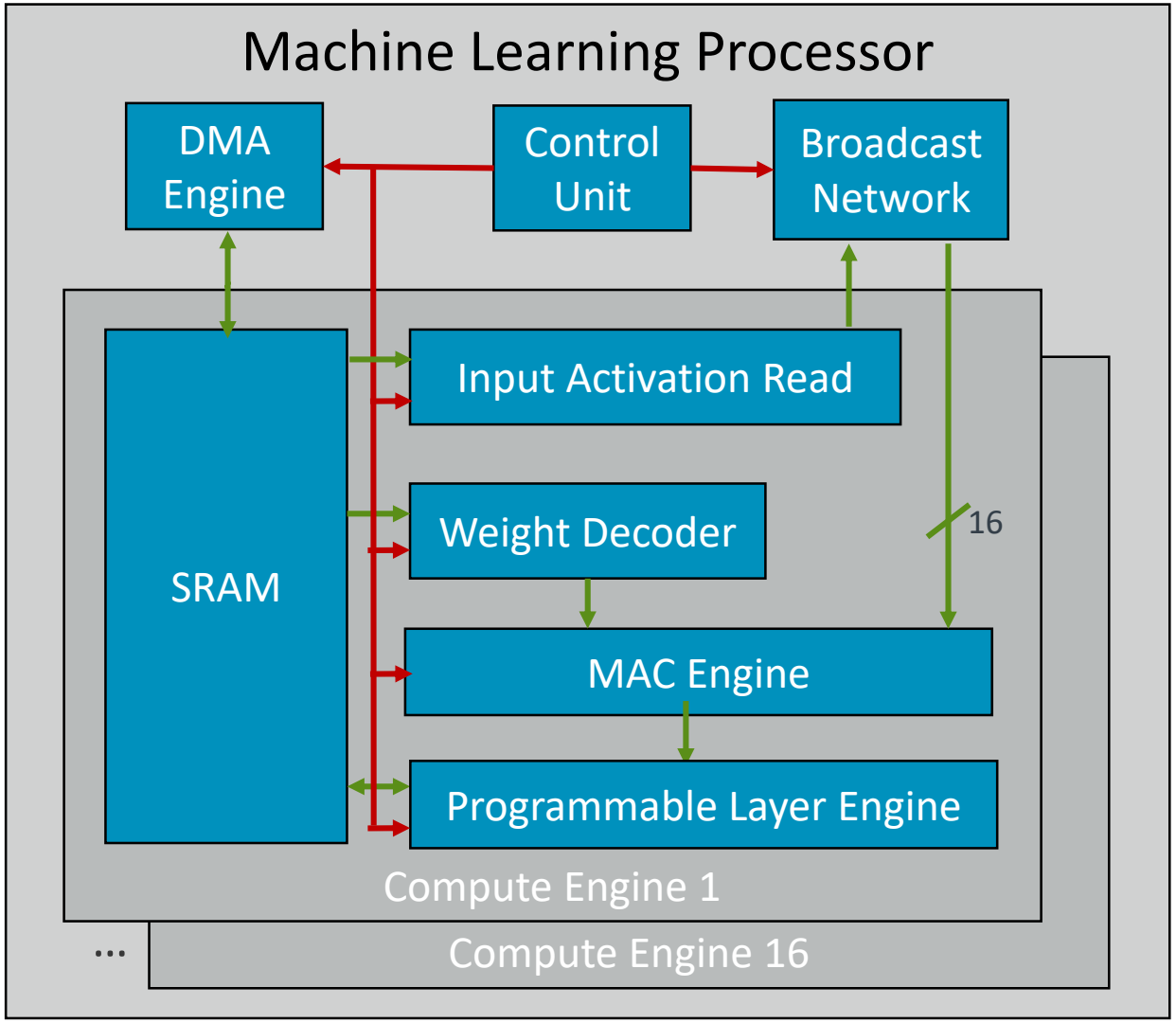
- ML processor supports
  - Weight Compression
  - Activation Compression
  - Tiling



arm

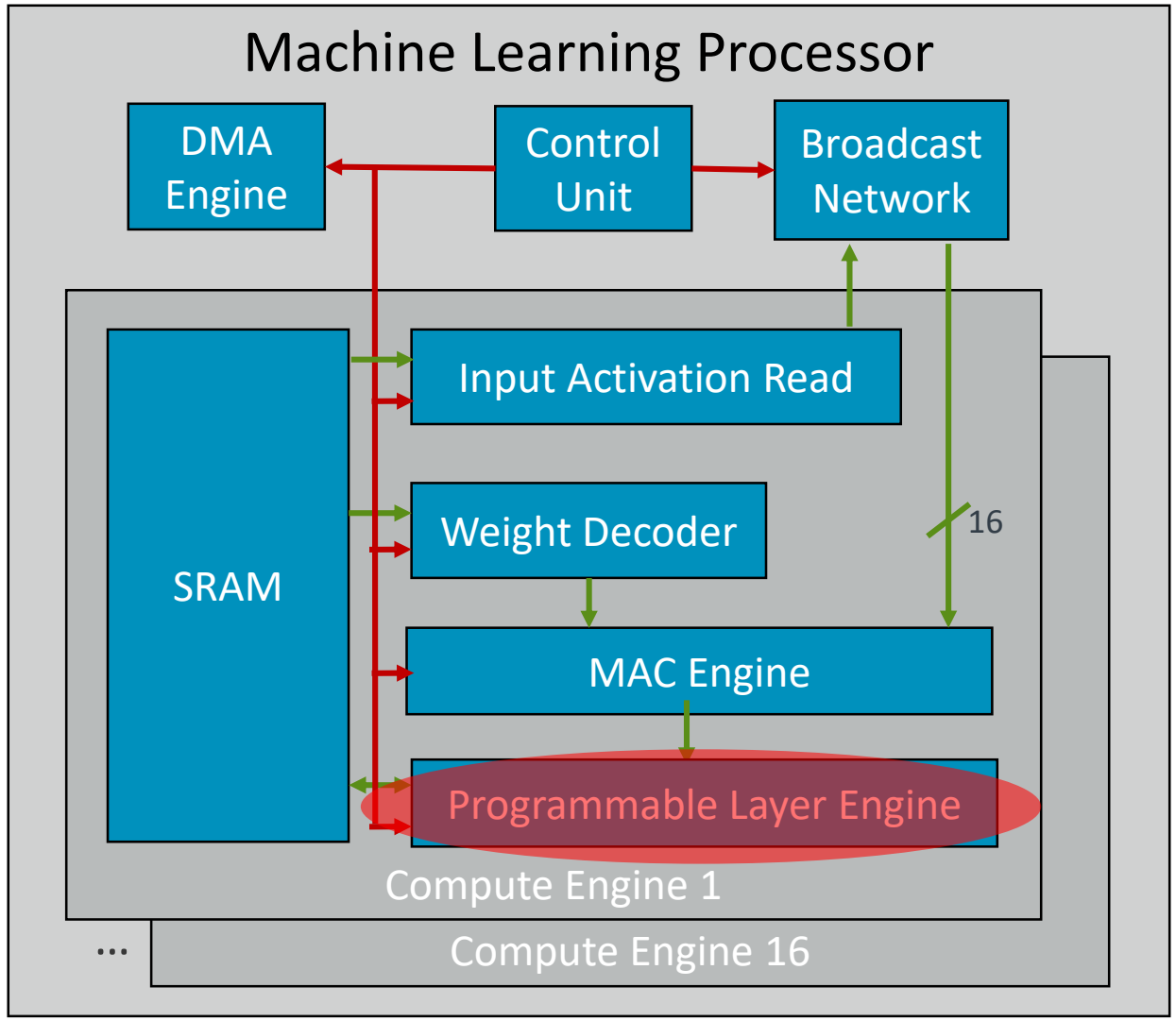
4 Key ingredients for a Machine Learning Processor

- Static scheduling
- Efficient convolutions
- Bandwidth reduction mechanisms
- Programmability/flexibility



arm

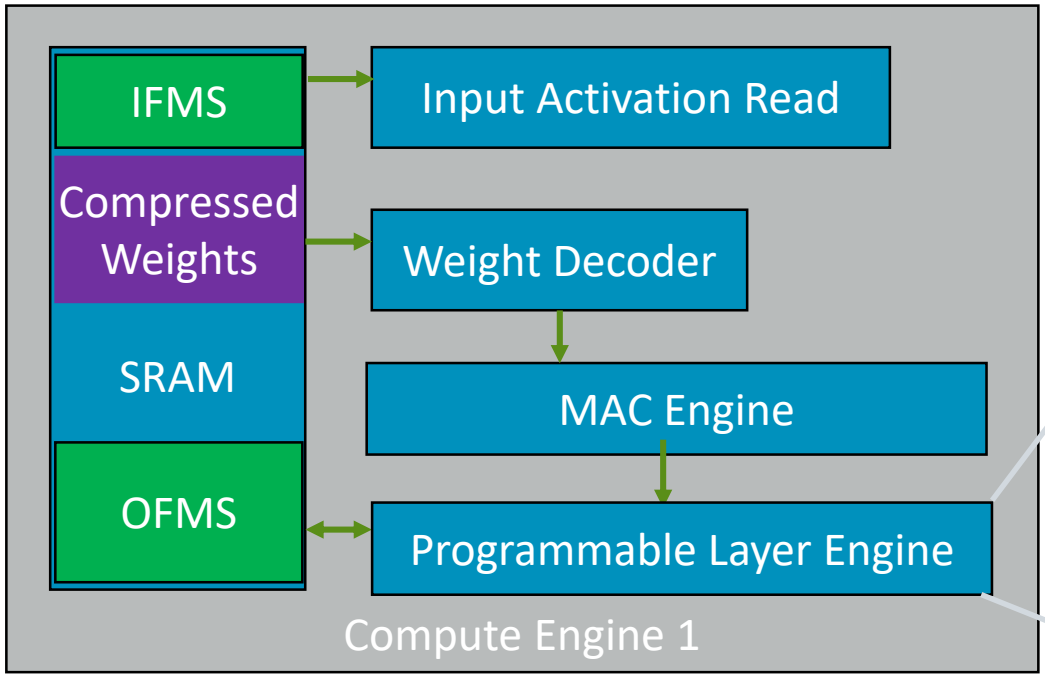
Programmable Layer Engine (PLE)

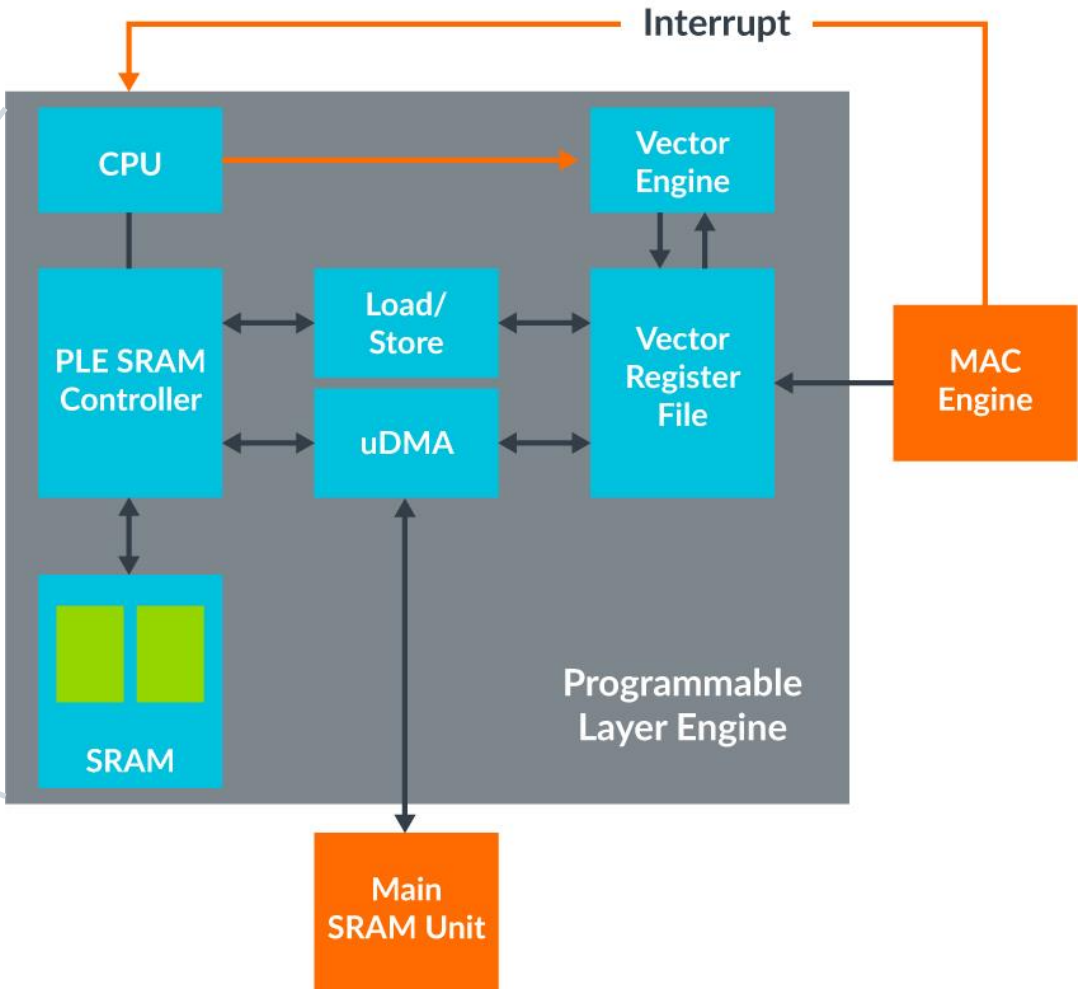


- State of the art in neural networks is still evolving
- Programmable Layer Engine
  - Provides design future-proofing
  - Benefits from existing Arm technology
- No hardware assumptions on operator ordering

arm

Programmable Layer Engine (PLE), cont.



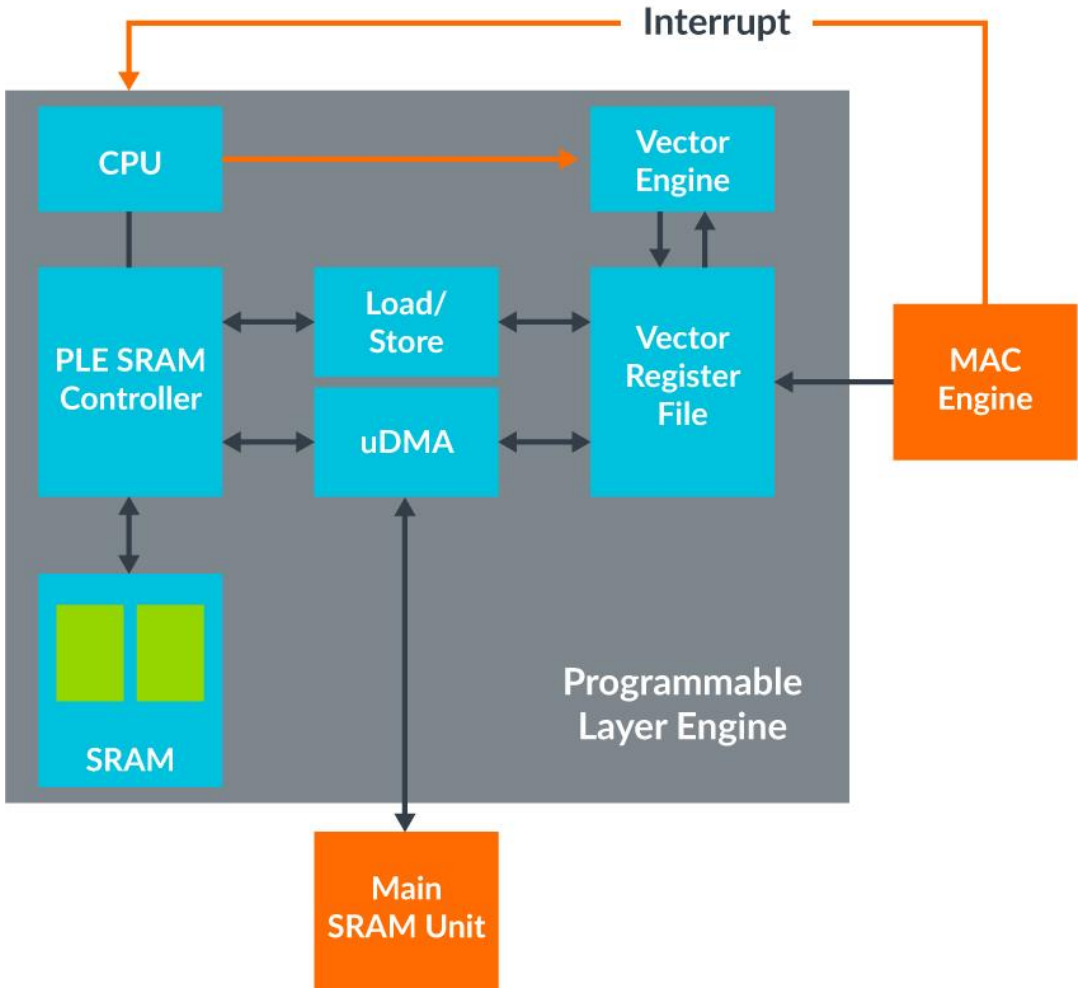


- We have extended Arm CPU technology with vector and NN extensions targeted for non-convolution operators (pooling, relu, etc)

arm

Programmable Layer Engine (PLE), cont.

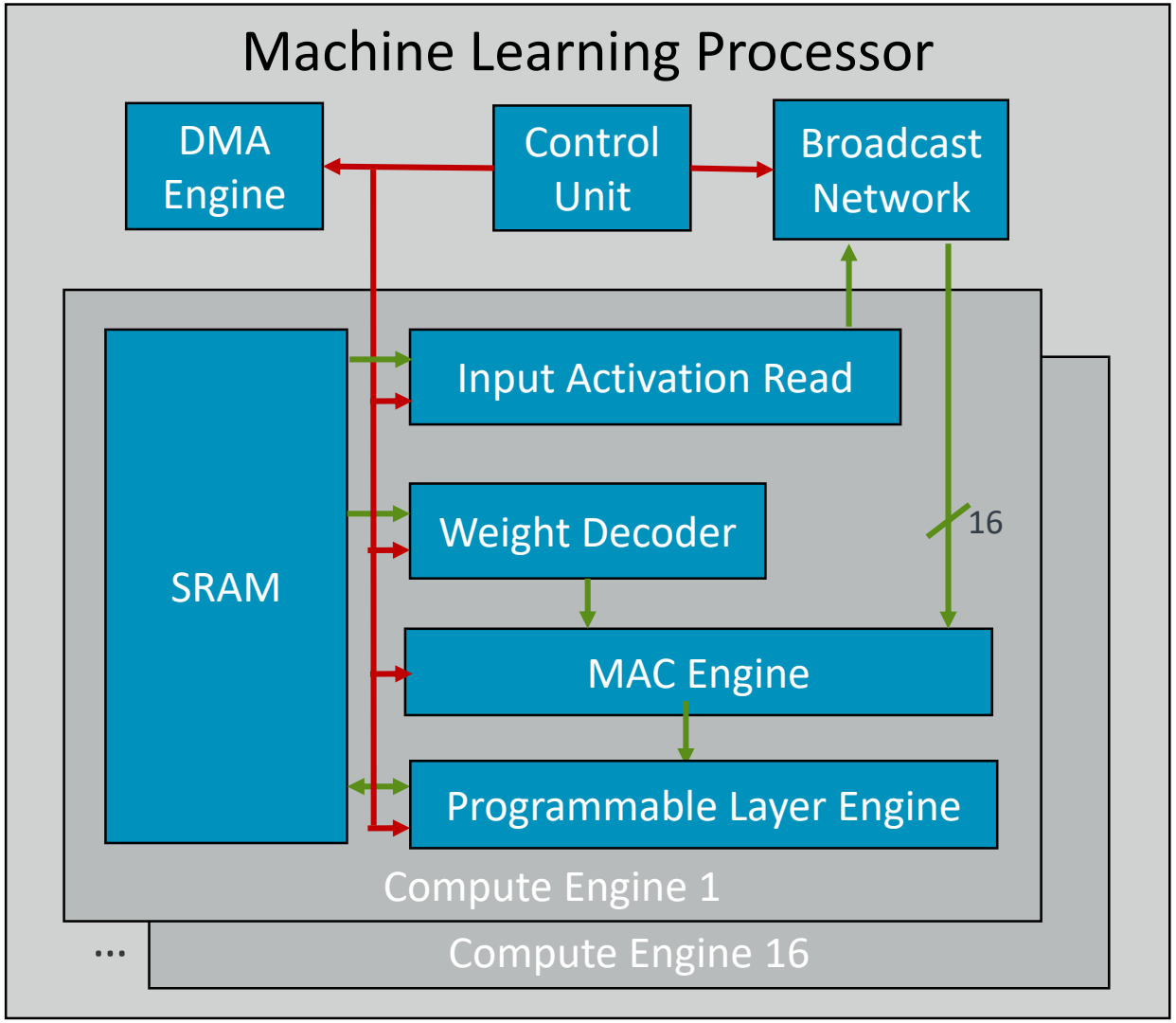
- The results of MAC computation are sent to the PLE
  - The PLE register file is populated directly
  - Interrupts are sent to activate PLE processing
  - The majority of operators are performed by a 16-lane vector engine – as they often pool or reduce
- Results are emitted back to SRAM
  - A micro-DMA unit writes data out
  - They are then fetched back into CE for subsequent processing



arm

Scalability

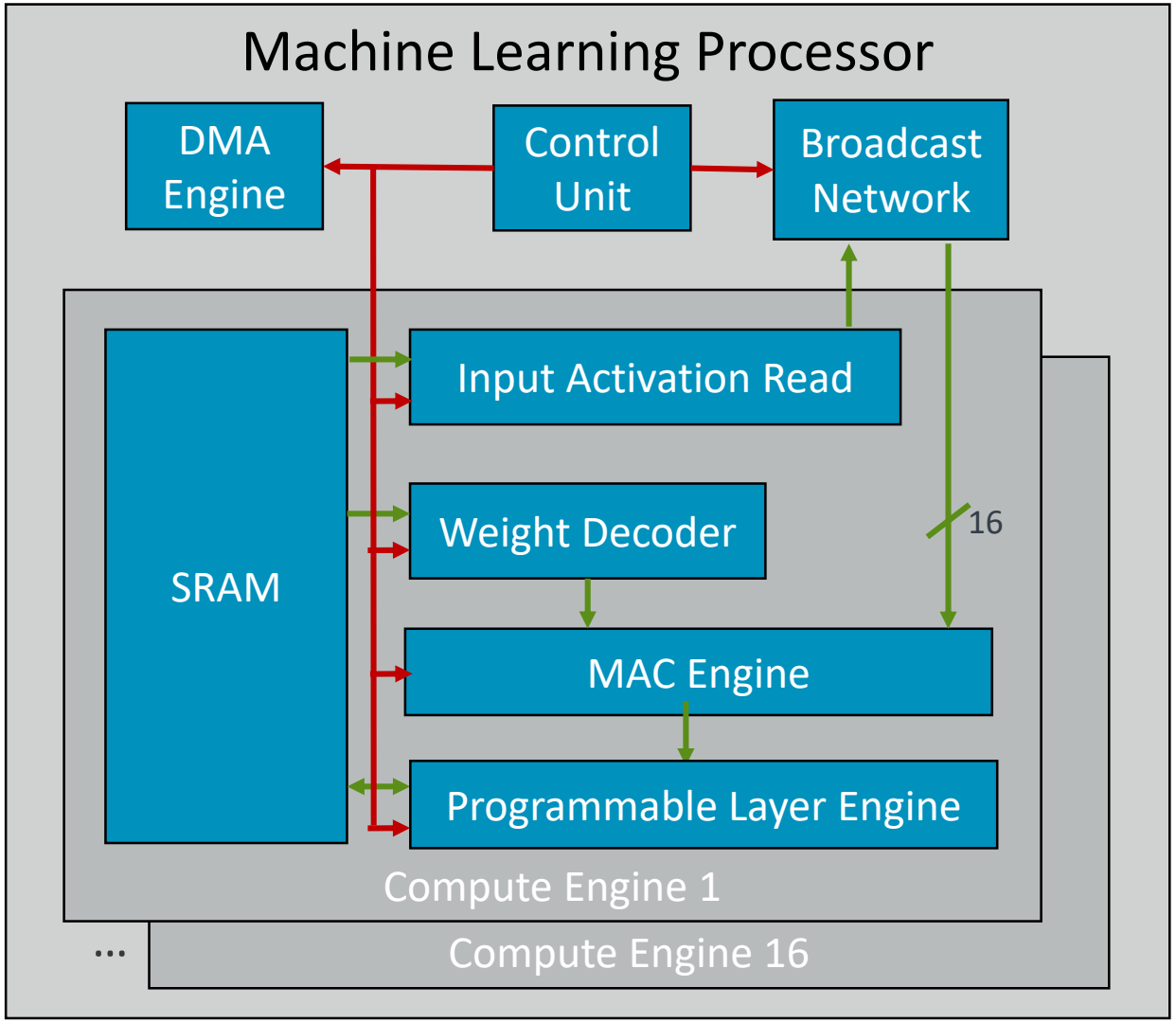
- Multiple ways to scale
  - Number of Compute Engines
  - MAC Engine throughput
  - Number of ML processors



arm

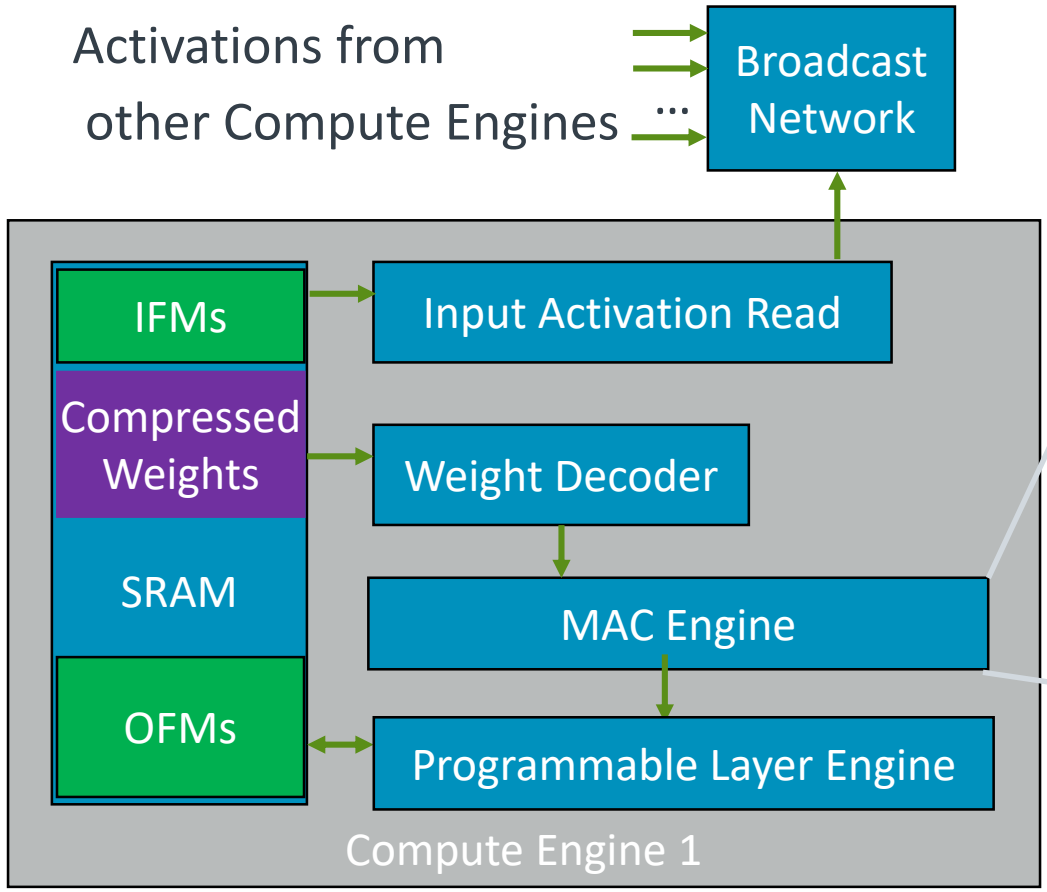
Arm's ML Processor

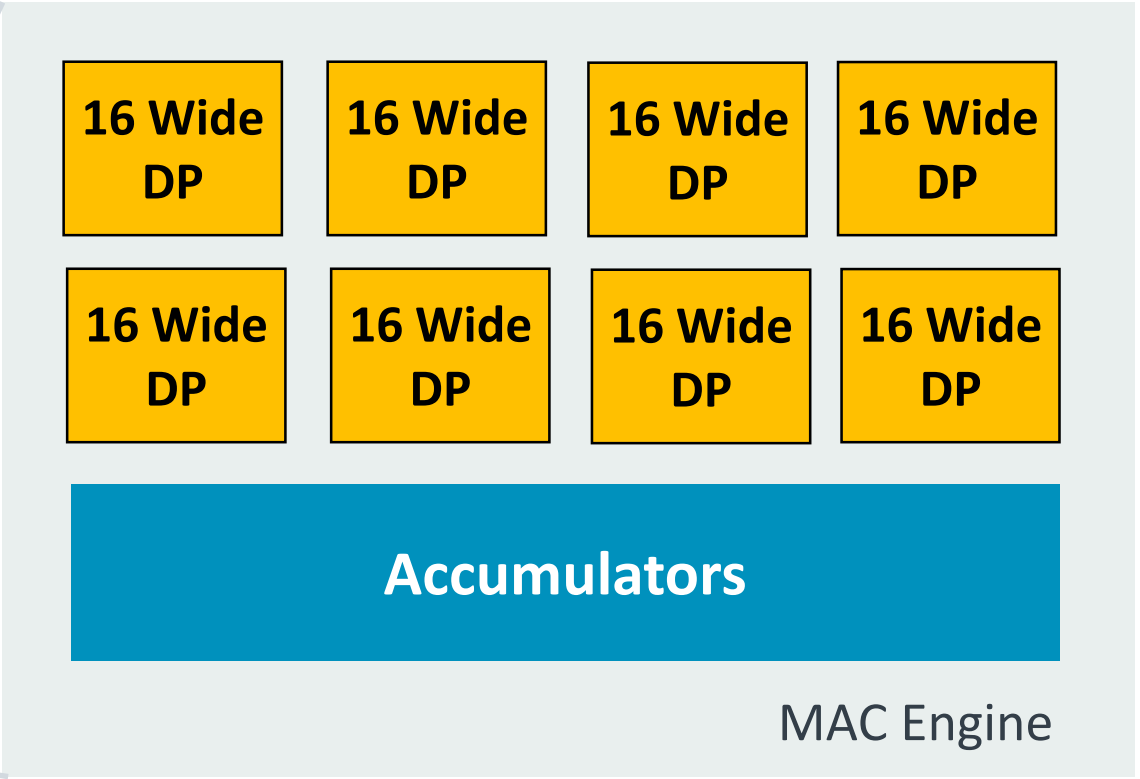
- 16 Compute Engines
- ~4 TOPS of convolution at 1 GHz.
- 8 bit quantized integer support
- 1MB of SRAM
- Support for Android NN API and Arm NN
- Optimized for CNNs, RNN support



Thank You
Danke
Merci
谢谢
ありがとう
Gracias
Kiitos
감사합니다
धन्यवाद
شكرًا
תודה

Convolutions

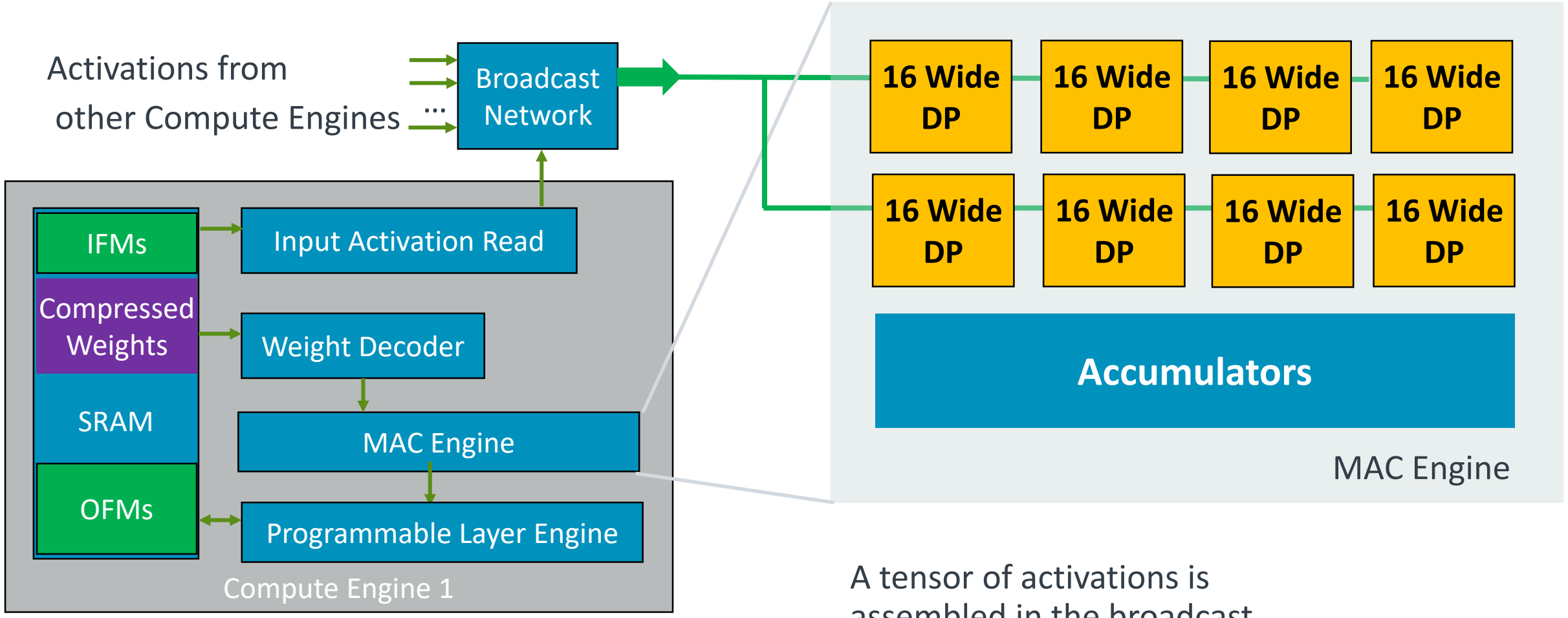




Each ”16 Wide DP” unit performs an 8b, 16 deep dot product operation

arm

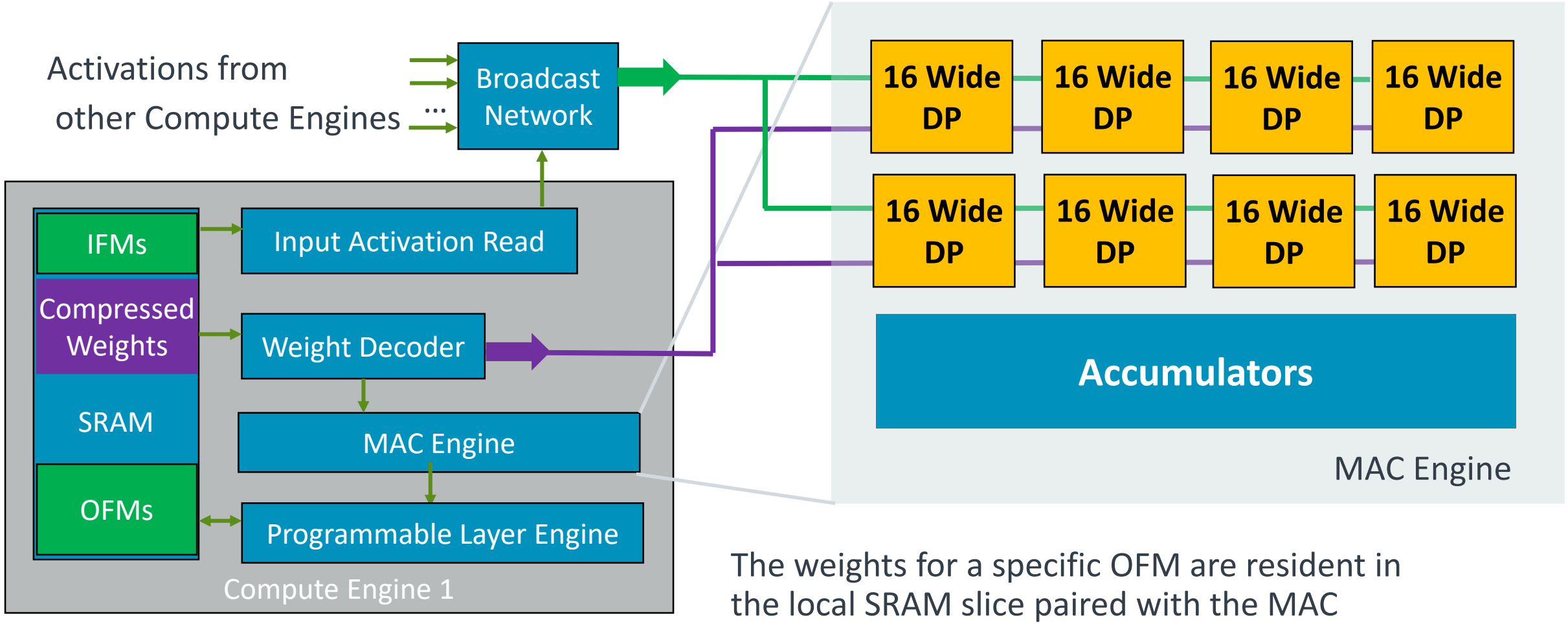
Convolutions



A tensor of activations is assembled in the broadcast network and sent to *all* MAC engines

arm

Convolutions

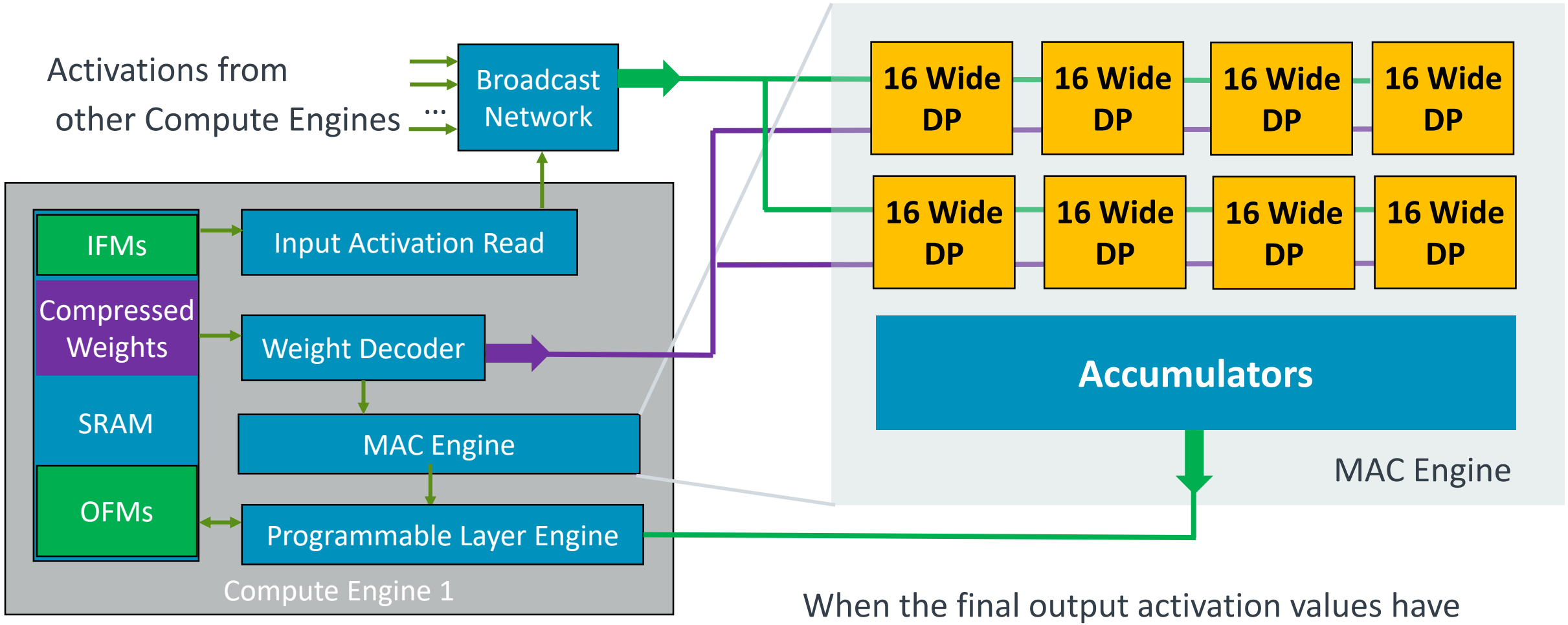


The weights for a specific OFM are resident in the local SRAM slice paired with the MAC Engine.

The weights are read, decompressed and sent to the MAC Engine

arm

Convolutions

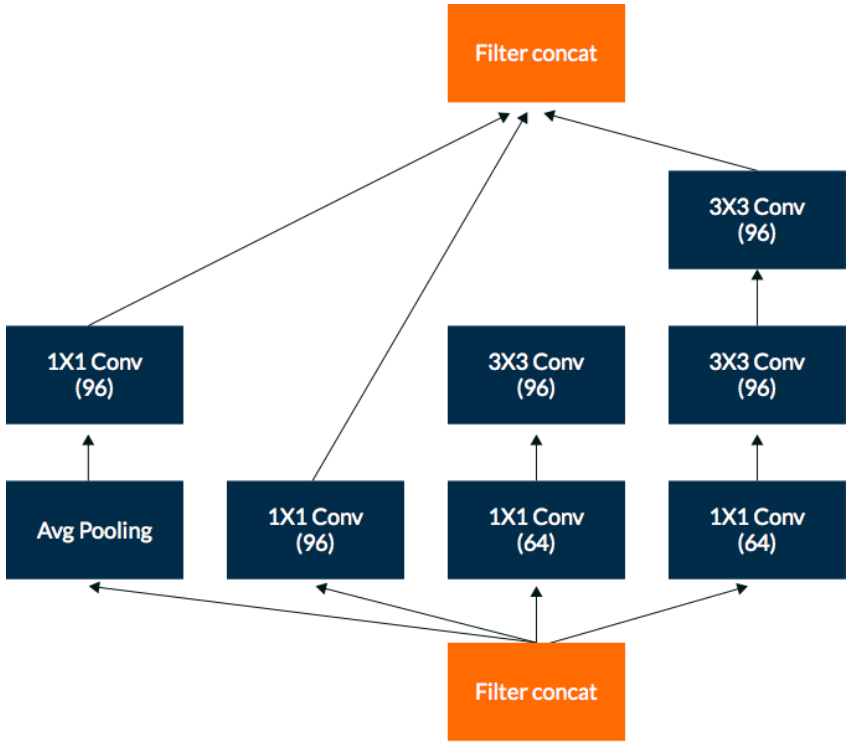


When the final output activation values have been communicated, the 32b values are scaled back to 8b and sent to the Programmable Layer Engine

arm

Tiling

- Compiler-based scheduling further reduces bandwidth
  - Scheduling tuned to keep working set in SRAM
  - Tiled or wide scheduling avoids trips to DRAM
  - Multiple outputs calculated in parallel from same input
  - Intermediate stages are pipelined between MAC and PLE
  - Possible because of static scheduling (compile time)



Szegedy et al
Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning
February 2016

arm