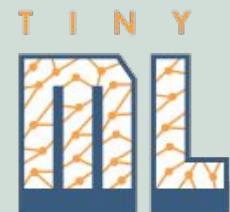


TinyML Challenges



Hardware



Software

Compute



Memory

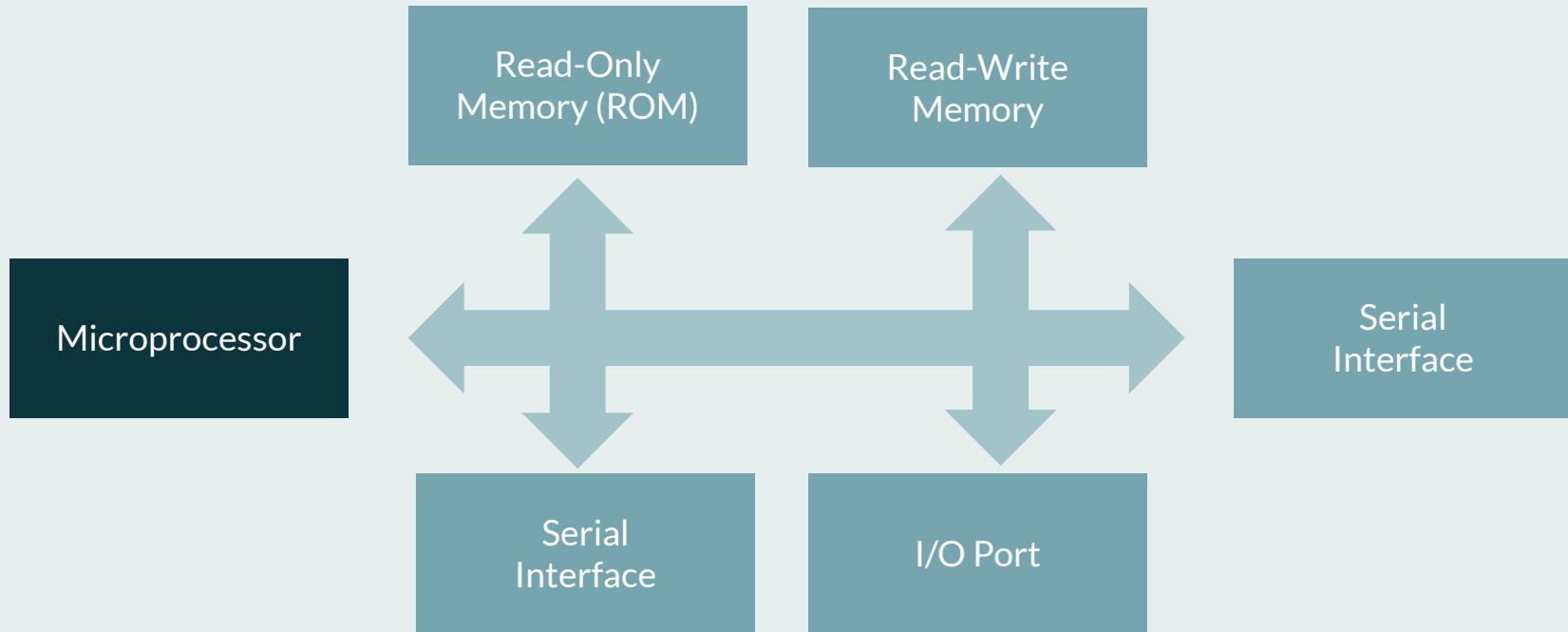


Storage



Microprocessor Vs Microcontroller

Microprocessor: only **one part** of the puzzle



Microcontroller



Microprocessor

- Heart of a **computer system**
- Just the processor, memory and storage are **external**
- Mainly used in general **purpose systems** like laptops, desktops and server
- **Offers flexibility** in design
- System size is **big**

Microcontroller

- Heart of an **embedded system**
- Memory and storage are all **internal** to the system
- Mainly used in **specialized, fixed function system** like phones, MP3 player, etc
- **Limited flexibility** in design
- System size is **tiny**

Orders of Magnitude Difference

	Microprocessor	>	Microcontroller
Platform			
Compute	1GHz - 4GHz	10x	1MHz - 400MHz
Memory	512MB - 64GB	10Kx	2KB - 512KB
Storage	64GB - 4TB	100Kx	32KB - 2MB
Power	30W - 100W	1Kx	150µW - 23.5mW

Microcontroller



1MHz - 400MHz

2KB - 512KB

32KB - 2MB

150 μ W - 23.5mW

Implications

- How complicated is the running task?
- How much memory does it need to have?
- How long does the job have to perform?

Hardware



Software

Software

Applications

Libraries

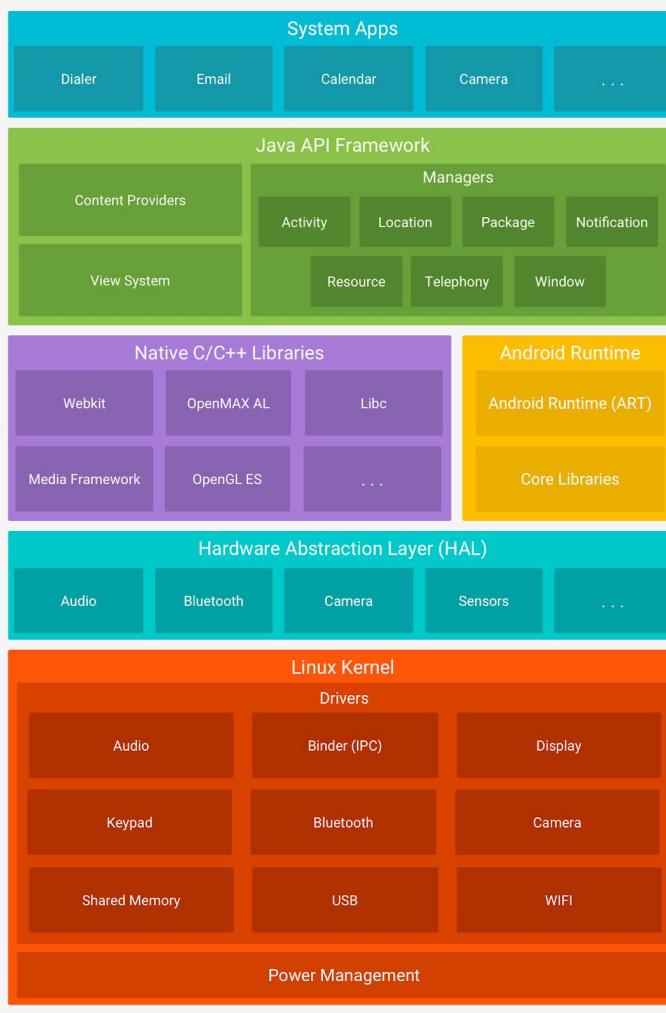
Operating System

Hardware

Widely Used Operating Systems



Mobile OS

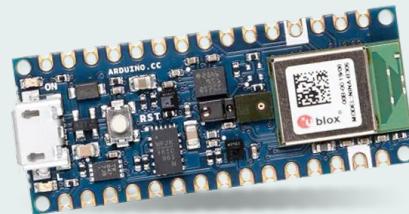


The Android Software Stack

Widely Used Operating Systems



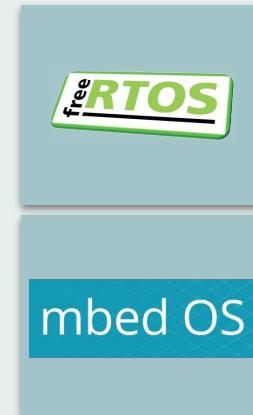
Embedded Systems



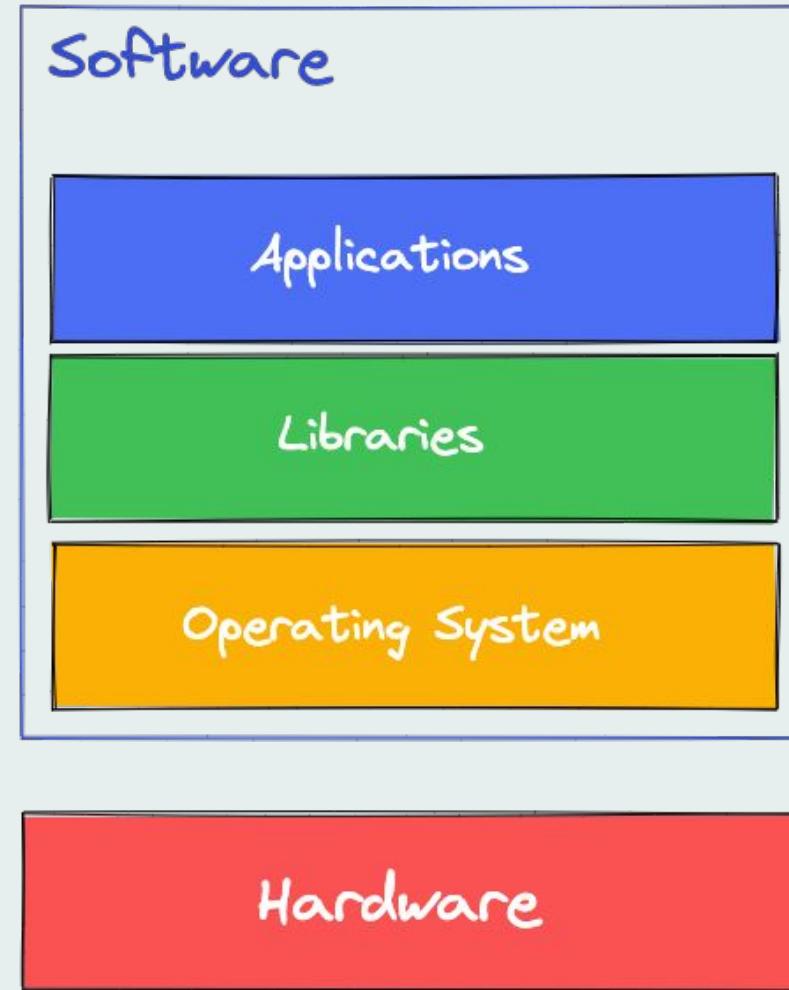
Widely Used Operating Systems



Mobile OS



Embedded Sys.



Software

Applications

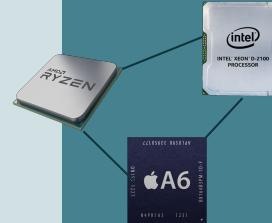
Libraries

Operating System

Hardware

```
import numpy as np
```

```
For x in range(10):  
    np.SaveTheWorld()
```



Software

Applications

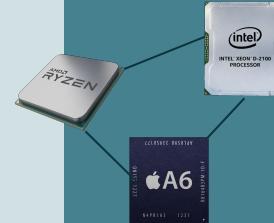
Libraries

Operating System

Hardware

Portability Opportunity

Able to execute the same code on different microprocessor hardware and architecture

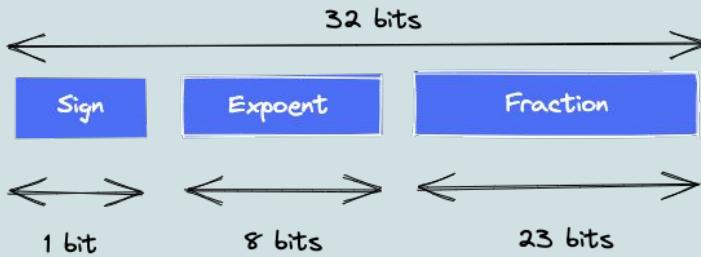


π

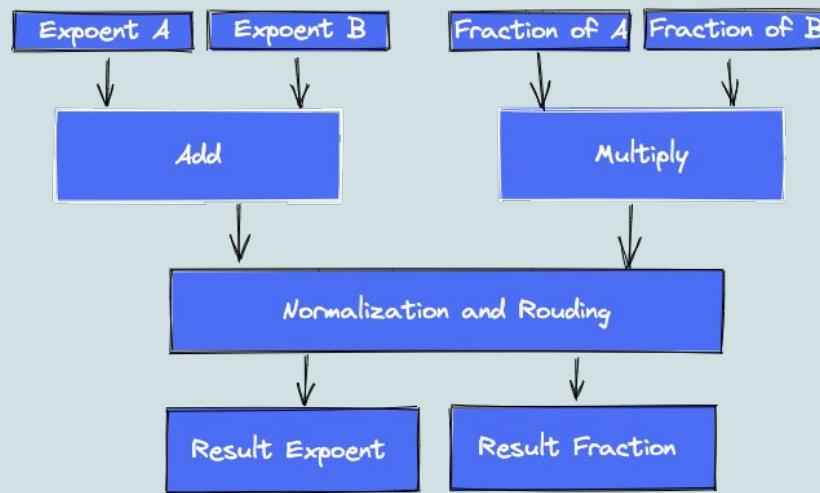
$$\frac{22}{7}$$

π

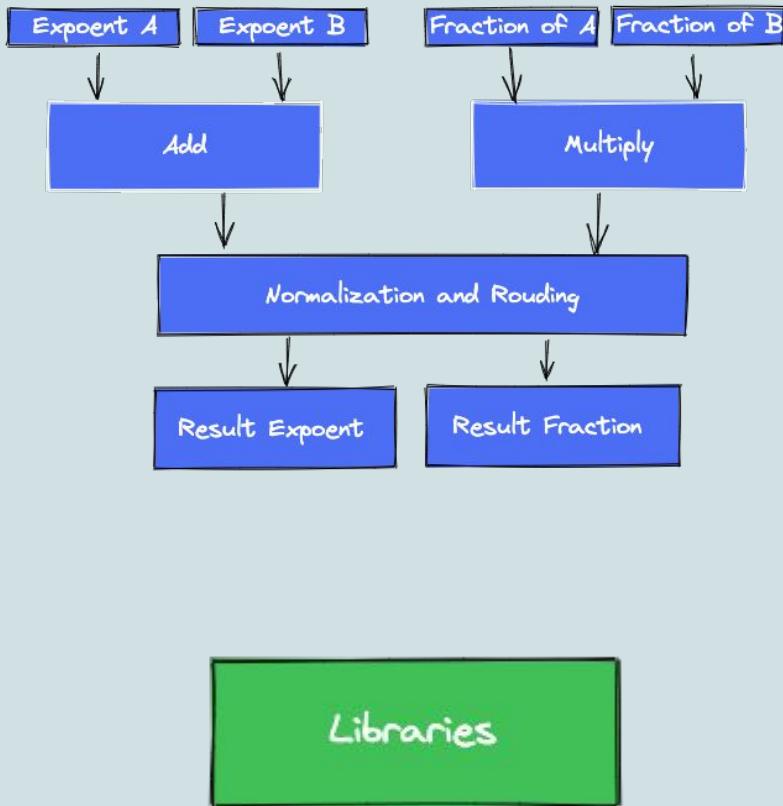
3.1415926535897932384626433832795028841971:
6939937510582097494459230781640628620899:
8628034825342117067982148086513282306647:
0938446095505822317253594081284811174502:
8410270193852110555964462294895493038196:
4428810975665933446128475648233786783165:
2712019091456485669234603486104543266482:
1339360726024914127372458700660631558817:
4881520920962829254091715364367892590360:
0113305305488204665213841469519415116094:
3305727036575959195309218611738193261179:
3105118548074462379962749567351885752724:
8912279381830119491298336733624406566430:
8602139494639522473719070217986094370277:
0539217176293176752384674818467669405132:
0005681271452635608277857713427577896091:
7363717872146844090122495343014654958537:
1050792279689258923542019956112129021960:
8640344181598136297747713099605187072113:
499999837297804995105973173281609631859:
5024459455346908302642522308253344685035:
2619311881710100031378387528865875332083:
8142061717766914730359825349042875546873:
1159562863882353787593751957781857780532:
171226806613001927876611195909216420199



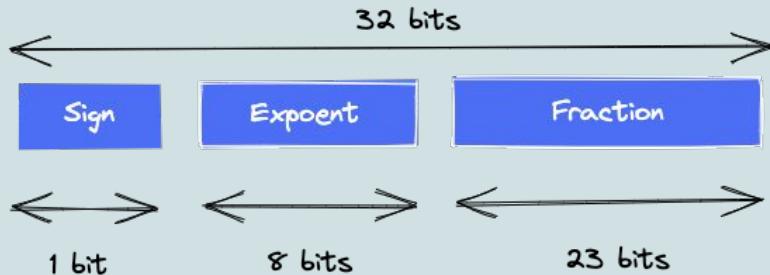
Single Precision
IEEE 754 Floating-Point Standard



3.1415926535897932384626433832795028841971:
 6939937510582097494459230781640628620899:
 8628034825342117067982148086513282306647:
 0938446095505822317253594081284811174502:
 8410270193852110555964462294895493038196:
 4428810975665933446128475648233786783165:
 2712019091456485669234603486104543266482:
 1339360726024914127372458700660631558817:
 4881520920962829254091715364367892590360:
 0113305305488204665213841469519415116094:
 3305727036575959195309218611738193261179:
 3105118548074462379962749567351885752724:
 8912279381830119491298336733624406566430:
 8602139494639522473719070217986094370277:
 0539217176293176752384674818467669405132:
 0005681271452635608277857713427577896091:
 7363717872146844090122495343014654958537:
 1050792279689258923542019956112129021960:
 8640344181598136297747713099605187072113:
 4999999837297804995105973173281609631859:
 5024459455346908302642522308253344685035:
 2619311881710100031378387528865875332083:
 8142061717766914730359825349042875546873:
 1159562863882353787593751957781857780532:
 171226806613001927876611195909216420199



3.1415926535897932384626433832795028841971:
6939937510582097494459230781640628620899:
8628034825342117067982148086513282306647:
0938446095505822317253594081284811174502:
8410270193852110555964462294895493038196:
4428810975665933446128475648233786783165:
2712019091456485669234603486104543266482:
1339360726024914127372458700660631558817:
4881520920962829254091715364367892590360:
0113305305488204665213841469519415116094:
3305727036575959195309218611738193261179:
3105118548074462379962749567351885752724:
8912279381830119491298336733624406566430:
8602139494639522473719070217986094370277:
0539217176293176752384674818467669405132:
0005681271452635608277857713427577896091:
7363717872146844090122495343014654958537:
1050792279689258923542019956112129021960:
8640344181598136297747713099605187072113:
499999837297804995105973173281609631859:
5024459455346908302642522308253344685035:
2619311881710100031378387528865875332083:
8142061717766914730359825349042875546873:
1159562863882353787593751957781857780532:
171226806613001927876611195909216420199

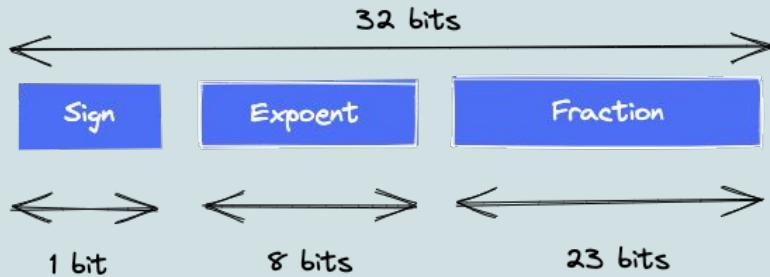


Single Precision
IEEE 754 Floating-Point Standard

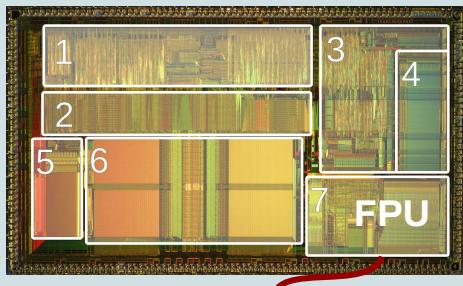


Hardware

3.1415926535897932384626433832795028841971:
 6939937510582097494459230781640628620899:
 8628034825342117067982148086513282306647:
 0938446095505822317253594081284811174502:
 8410270193852110555964462294895493038196:
 4428810975665933446128475648233786783165:
 2712019091456485669234603486104543266482:
 1339360726024914127372458700660631558817:
 4881520920962829254091715364367892590360:
 0113305305488204665213841469519415116094:
 3305727036575959195309218611738193261179:
 3105118548074462379962749567351885752724:
 8912279381830119491298336733624406566430:
 8602139494639522473719070217986094370277:
 0539217176293176752384674818467669405132:
 0005681271452635608277857713427577896091:
 7363717872146844090122495343014654958537:
 1050792279689258923542019956112129021960:
 8640344181598136297747713099605187072113:
 499999837297804995105973173281609631859:
 5024459455346908302642522308253344685035:
 2619311881710100031378387528865875332083:
 8142061717766914730359825349042875546873:
 1159562863882353787593751957781857780532:
 171226806613001927876611195909216420199



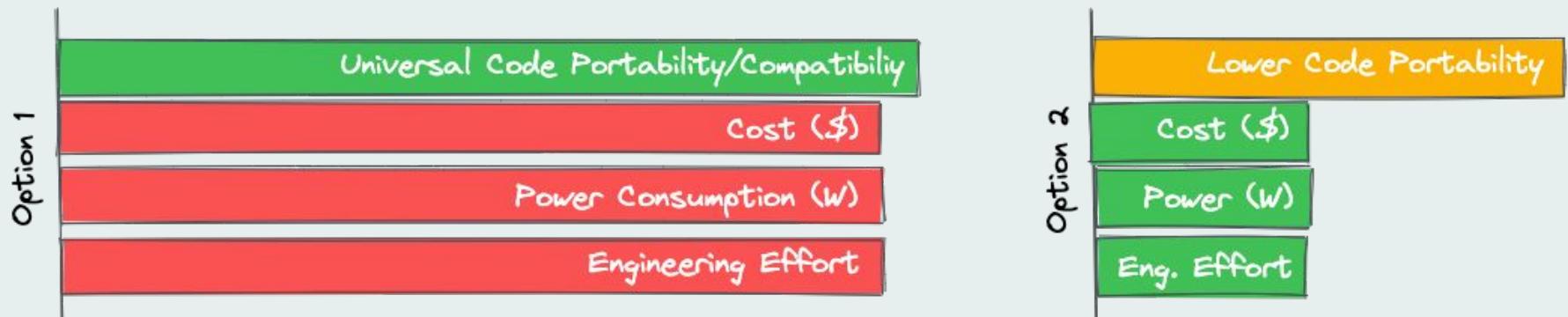
Single Precision
IEEE 754 Floating-Point Standard



Hardware

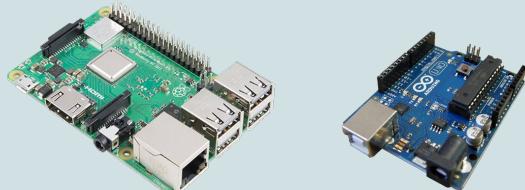
3.1415926535897932384626433832795028841971:
6939937510582097494459230781640628620899:
8628034825342117067982148086513282306647:
0938446095505822317253594081284811174502:
8410270193852110555964462294895493038196:
4428810975665933446128475648233786783165:
2712019091456485669234603486104543266482:
1339360726024914127372458700660631558817:
4881520920962829254091715364367892590360:
0113305305488204665213841469519415116094:
3305727036575959195309218611738193261179:
3105118548074462379962749567351885752724:
8912279381830119491298336733624406566430:
8602139494639522473719070217986094370277:
0539217176293176752384674818467669405132:
0005681271452635608277857713427577896091:
7363717872146844090122495343014654958537:
1050792279689258923542019956112129021960:
8640344181598136297747713099605187072113:
499999837297804995105973173281609631859:
5024459455346908302642522308253344685035:
2619311881710100031378387528865875332083:
8142061717766914730359825349042875546873:
1159562863882353787593751957781857780532:
171226806613001927876611195909216420199

Portability Trade-offs



Portability Trade-offs

Sacrifice portability across systems for efficiency in system performance and power efficiency

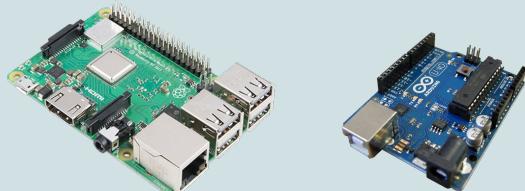


Specific HW Implementation of a Library



Portability Trade-offs

Sacrifice portability across systems for efficiency in system performance and power efficiency



Specific HW Implementation of a Library

Question:

How do we enable TinyML uniformly across these different systems if there is lower platform portability?

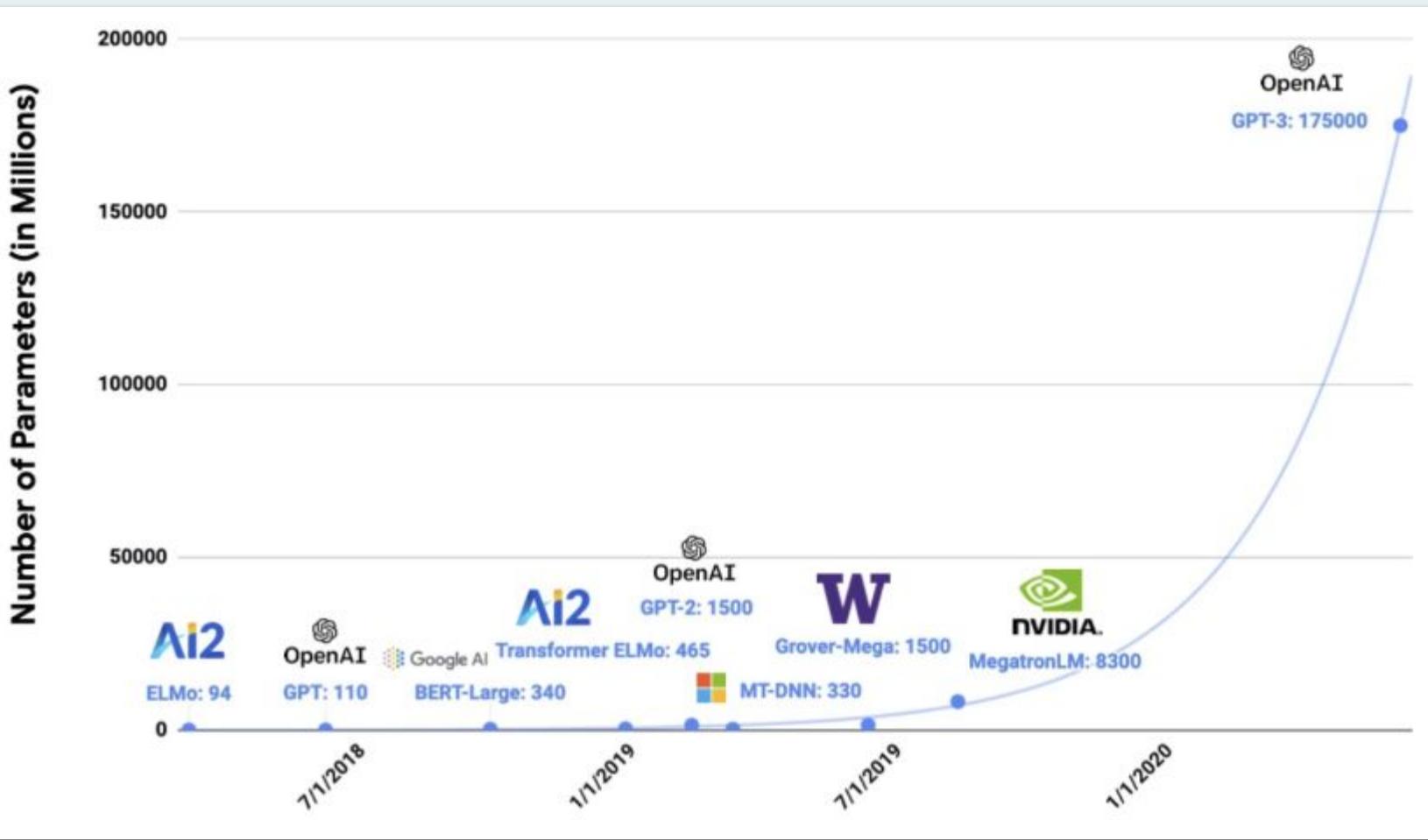
Summary

Embedded hardware is extremely limited in performance, power consumption and storage

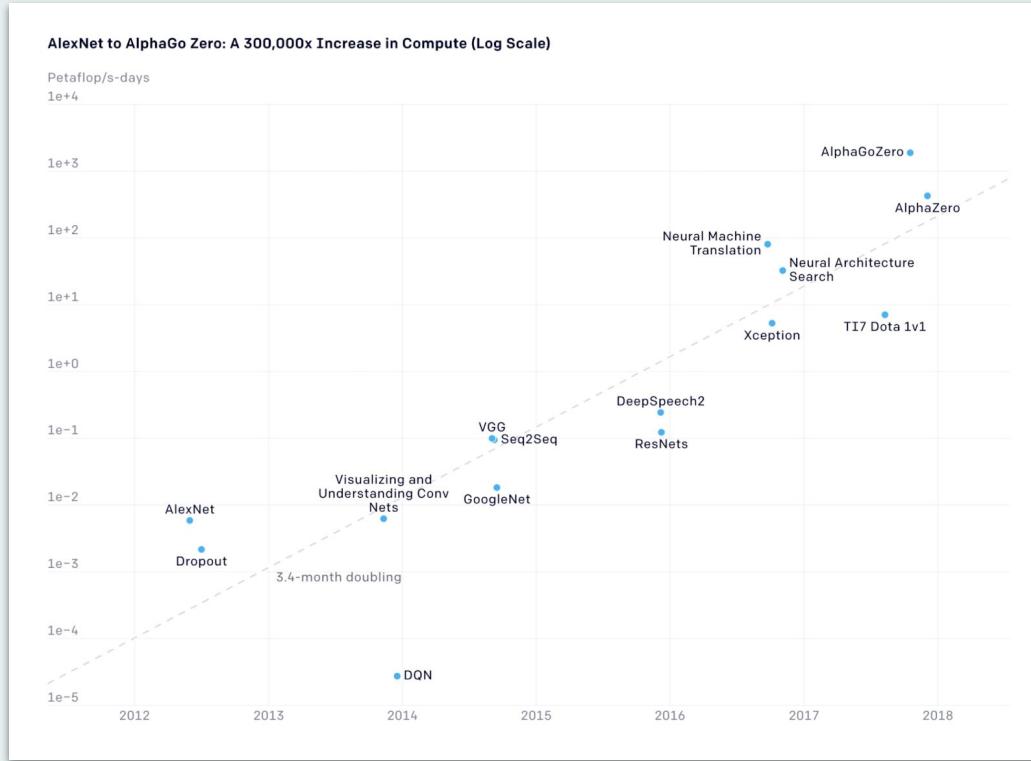
Embedded software is not as portable and flexible as mainstream computing



TinyML

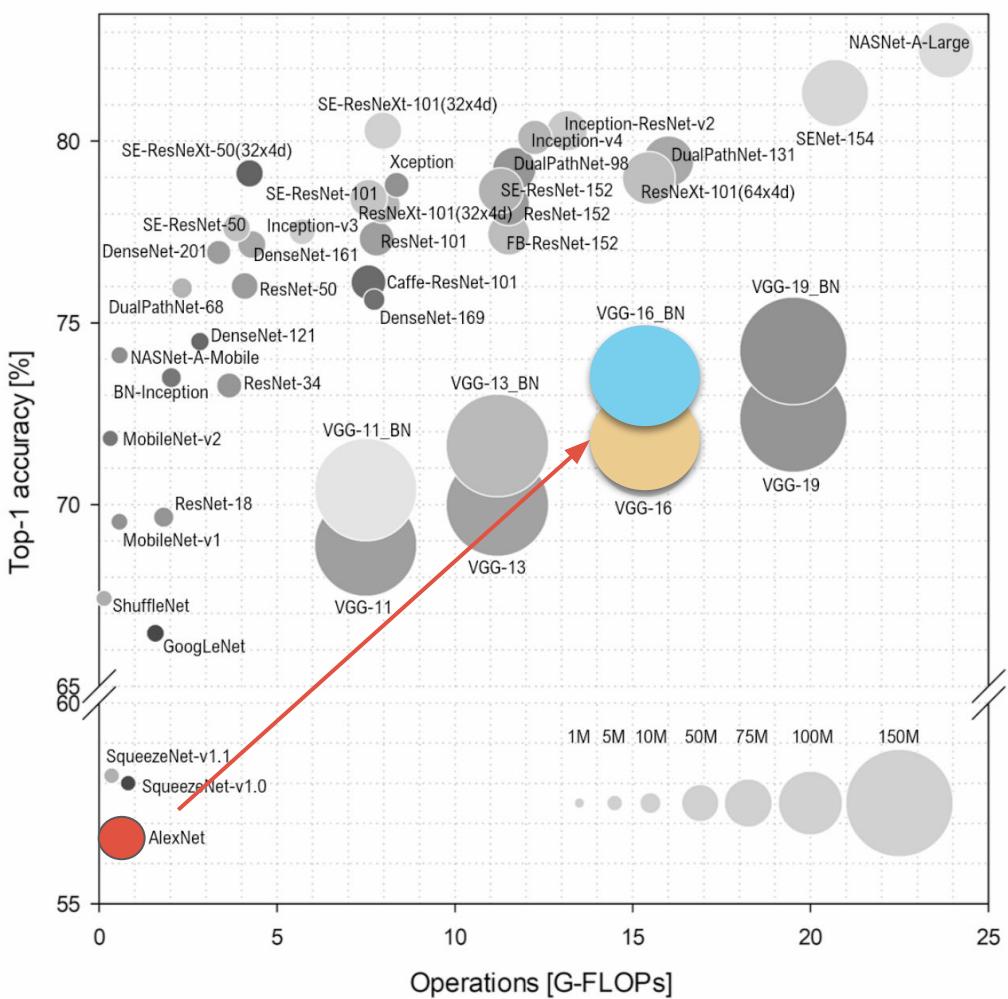


ML Compute Needs (2012 to Present Day)



In recent years,
**computing needs grew
by 300,000x** to train
the machine learning
models that are widely
deployed in the
industry

ML Model Evolution

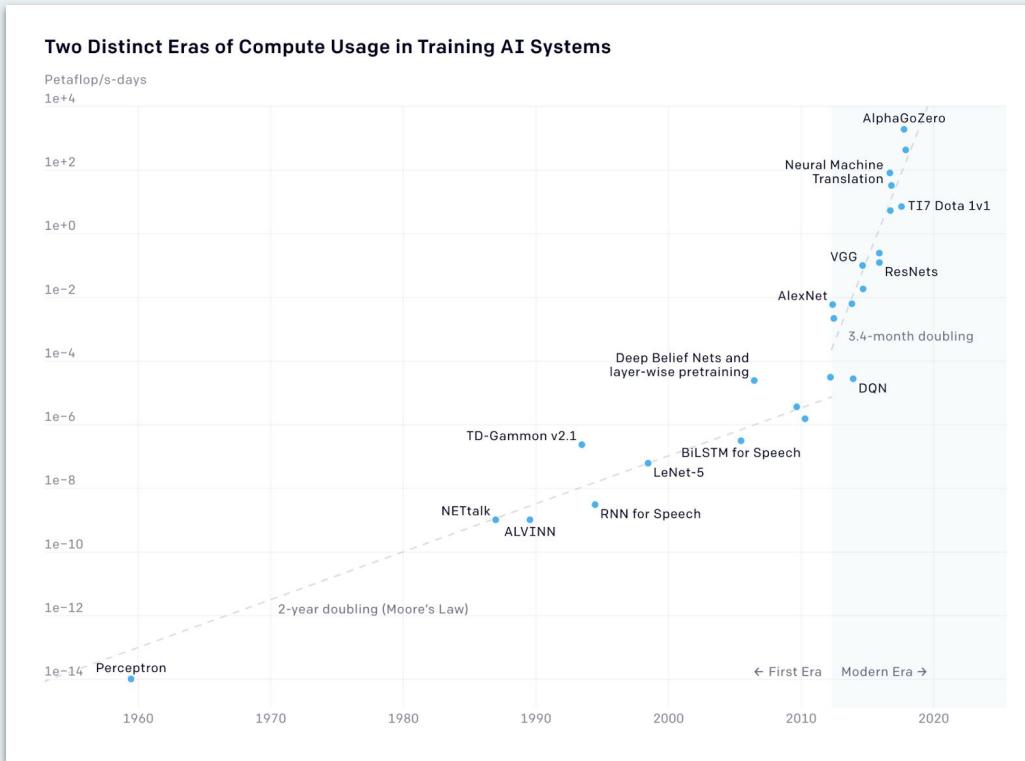


- VGGNet (2014)
 - 71.5% accuracy
 - 528 MB in size

Source: S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures". IEEE Access, vol. 6, 2018.



ML Compute Needs (from the 1960s)



In recent years, the amount of computing needed has grown remarkably fast.

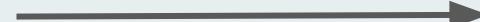
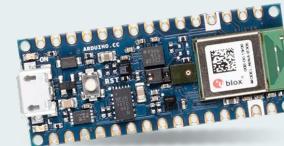
Computer requirements are **doubling nearly every 3 to 4 months.**



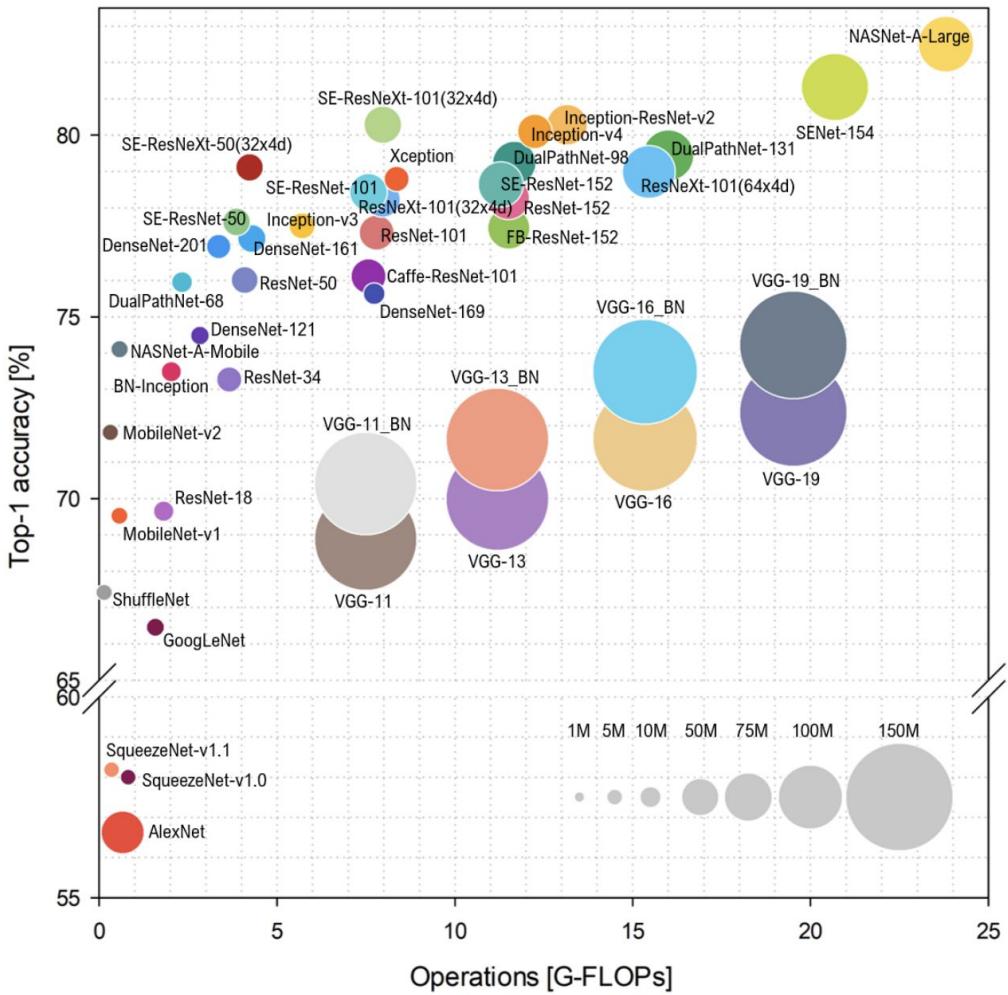
Cloud TPU



TinyML

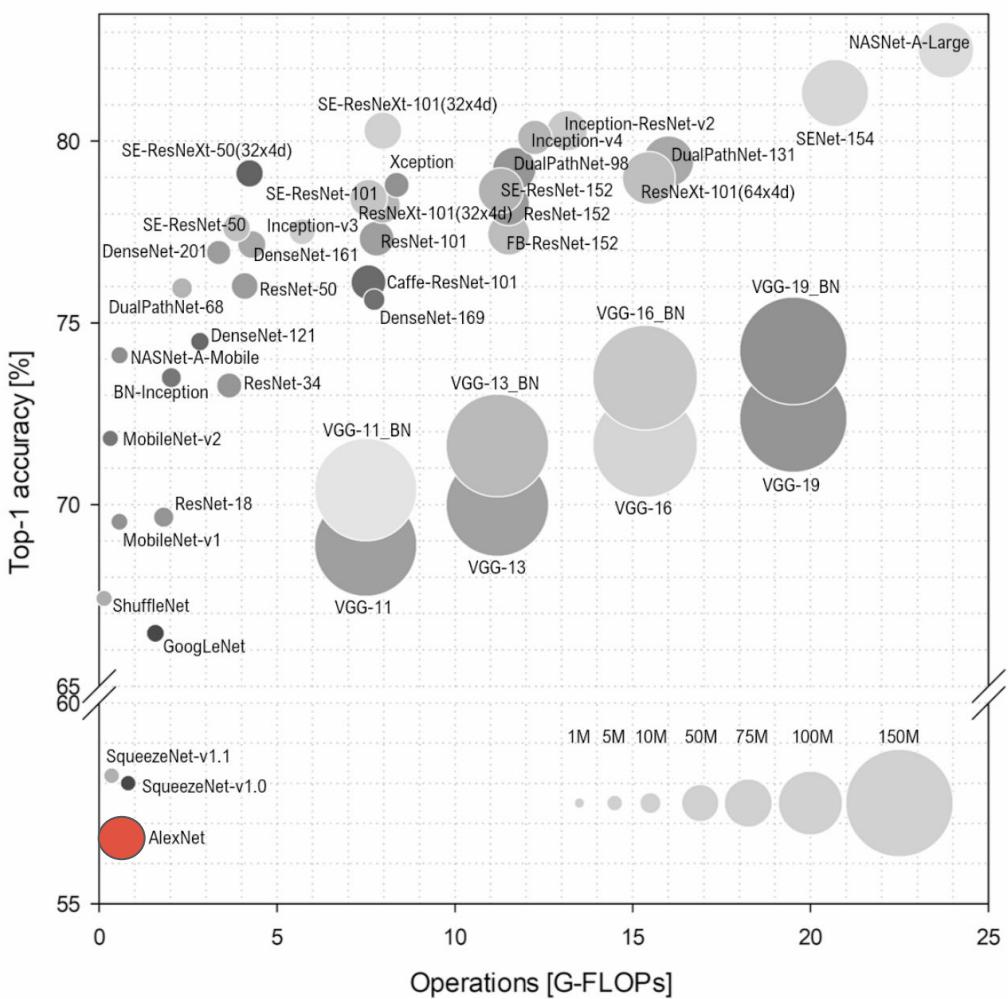


ML Model Evolution



Source: S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures". IEEE Access, vol. 6, 2018.

ML Model Evolution

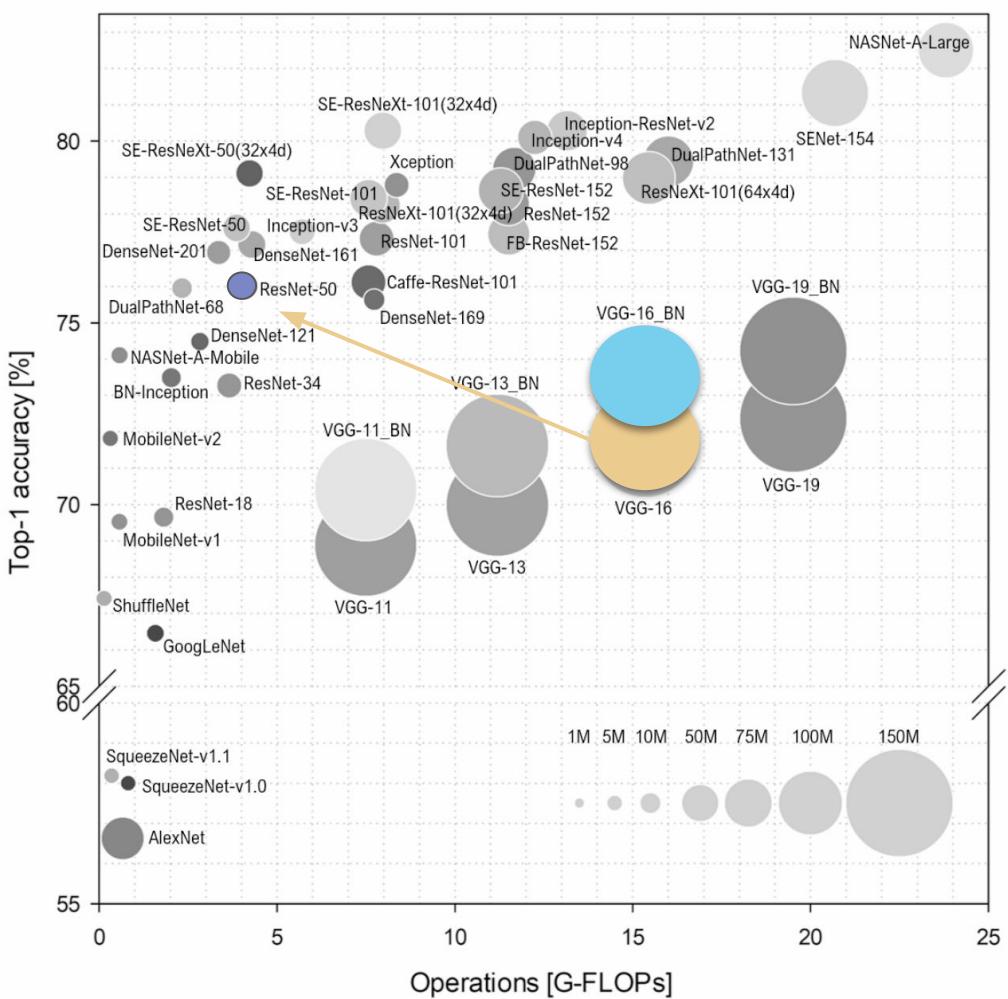


- AlexNet (2012)
 - 57.1% accuracy
 - 61 MB in size

Source: S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures". IEEE Access, vol. 6, 2018.



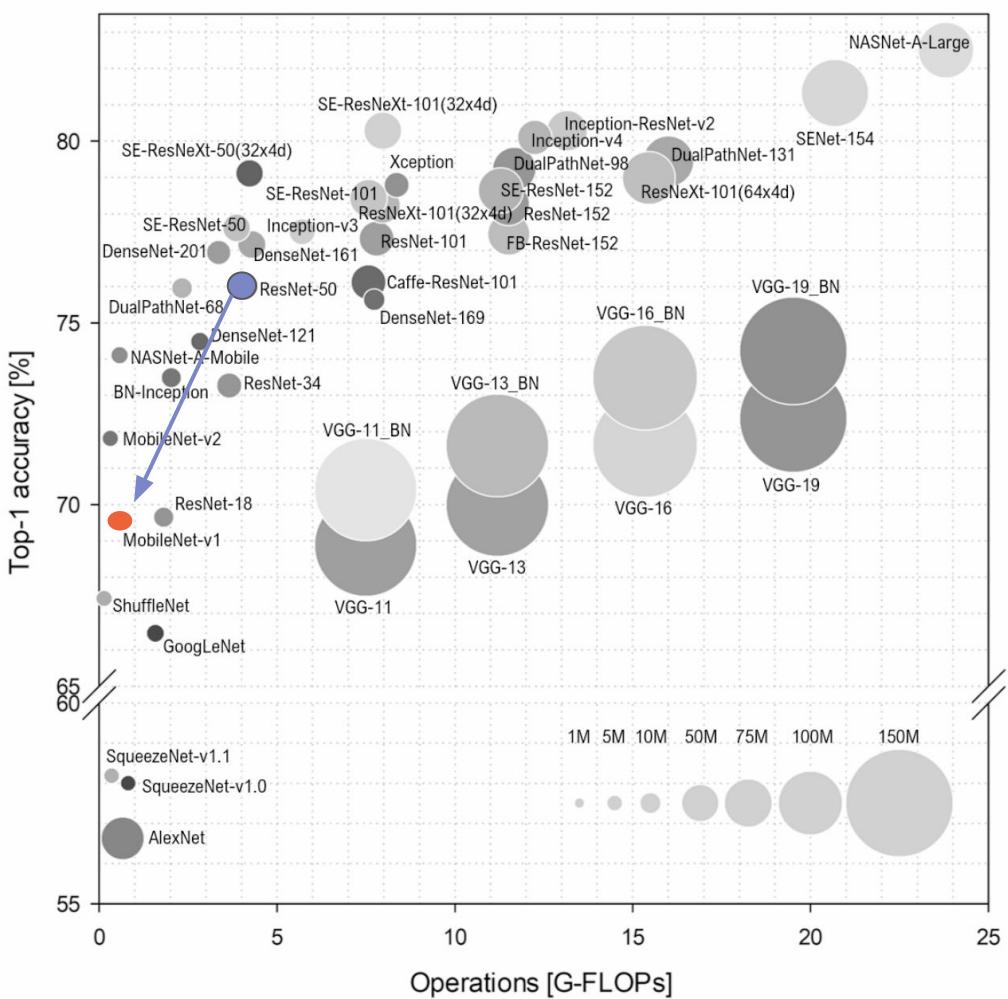
ML Model Evolution



- ResNet (2015)
 - 75.8% accuracy
 - 22.7 MB in size

Source: S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures". IEEE Access, vol. 6, 2018.

ML Model Evolution



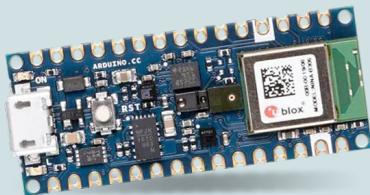
- **MobileNet (2015)**
 - 70.6% accuracy
 - 16.9 MB in size

Source: S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures". IEEE Access, vol. 6, 2018.



Problem:

Our board only has **256 KB** of RAM (memory) yet MobileNetv1 needs **16.9MB!!!**



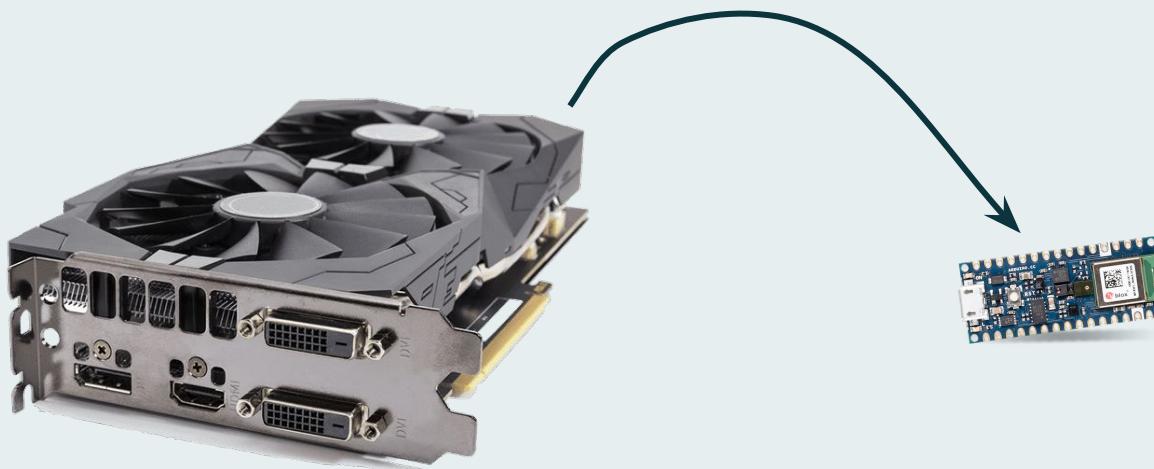
ML Model Evolution

MobileNet (**2015**)

- 70.6% accuracy
- 16.9 MB in size

Source: S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures". IEEE Access, vol. 6, 2018.

What are the Challenges for TinyML?



Machine Learning Models



Machine Learning Runtimes

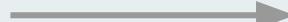
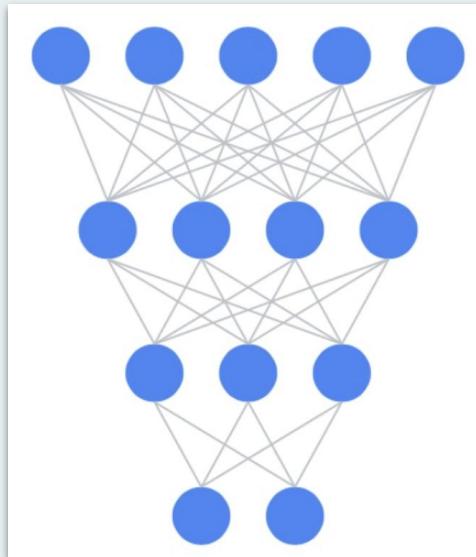


Machine Learning Hardware

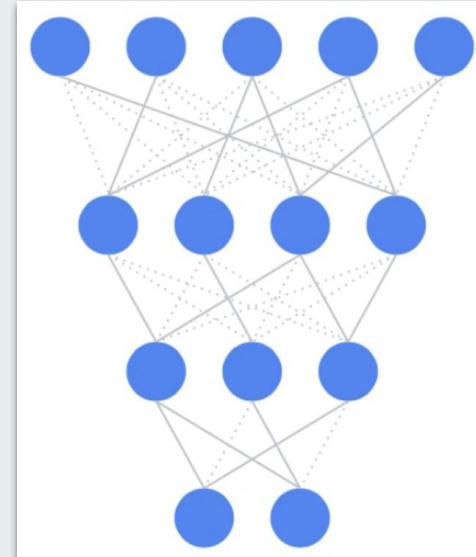
Model Compression Techniques

Pruning
Quantization
Knowledge Distillation
...

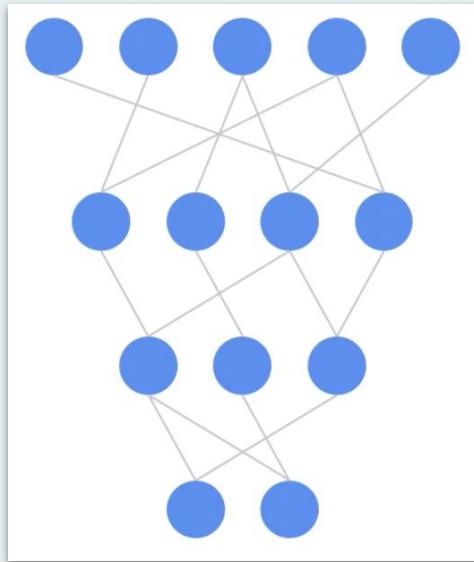
Pruning



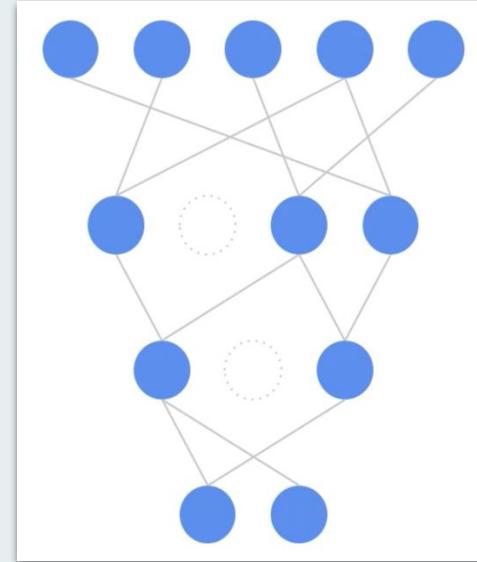
Pruning
Synapses



Pruning



→
Pruning
Neurons



Machine Learning Models



Machine Learning Runtimes

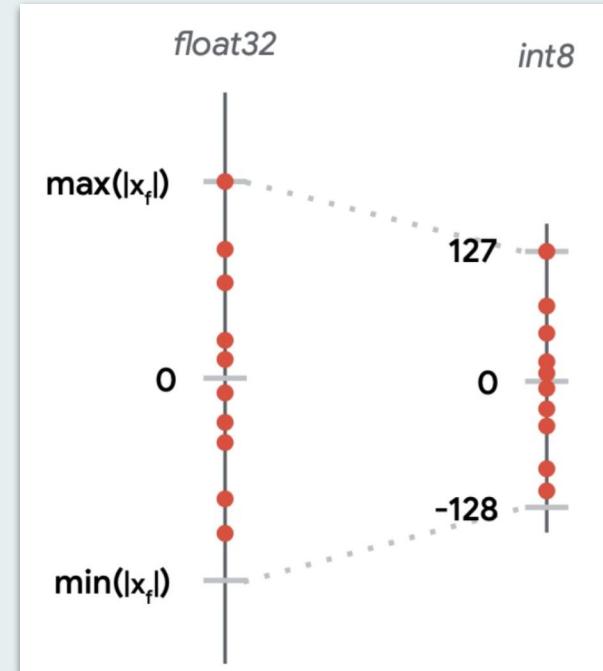


Machine Learning Hardware

Model Compression Techniques

Pruning
Quantization
Knowledge Distillation
...

Quantization



Machine Learning Models



Machine Learning Runtimes



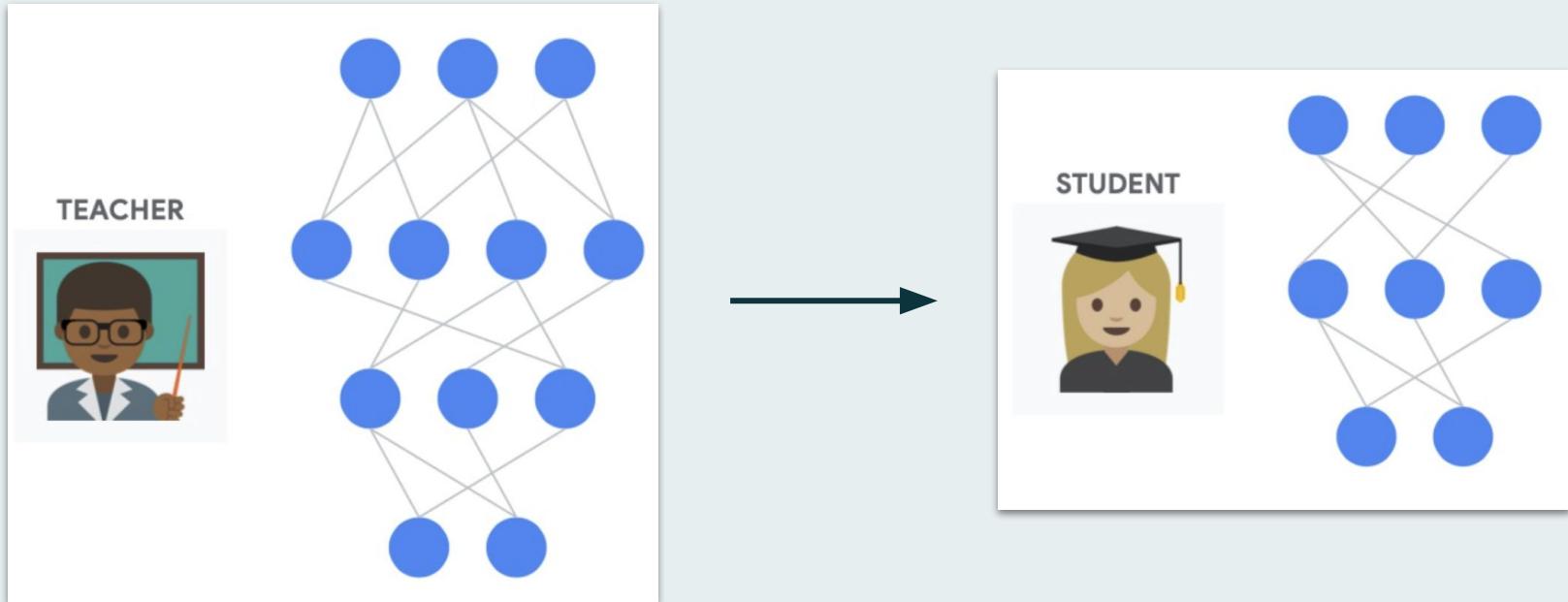
Machine Learning Hardware

Model Compression Techniques

Pruning
Quantization
Knowledge Distillation

...

Knowledge Distillation



Machine Learning Models



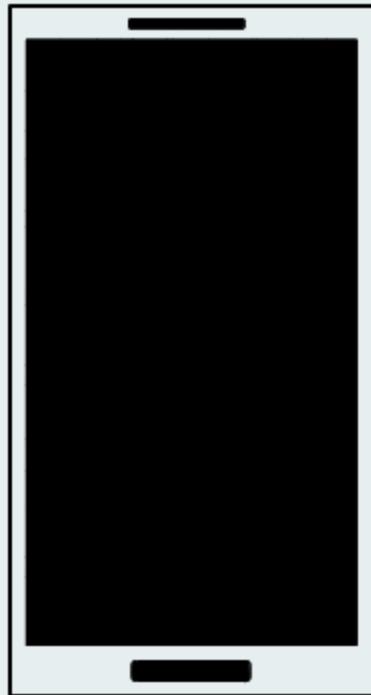
Machine Learning Runtimes



Machine Learning Hardware



TensorFlow



- Less Memory
- Less computer power
- Only focused on inference

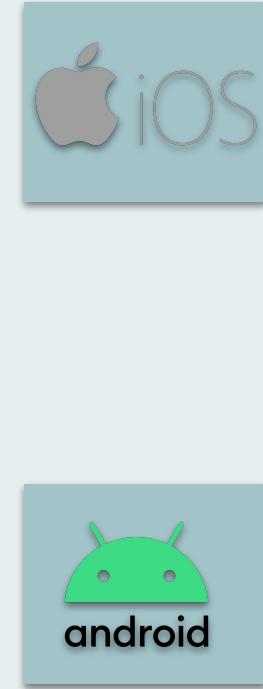
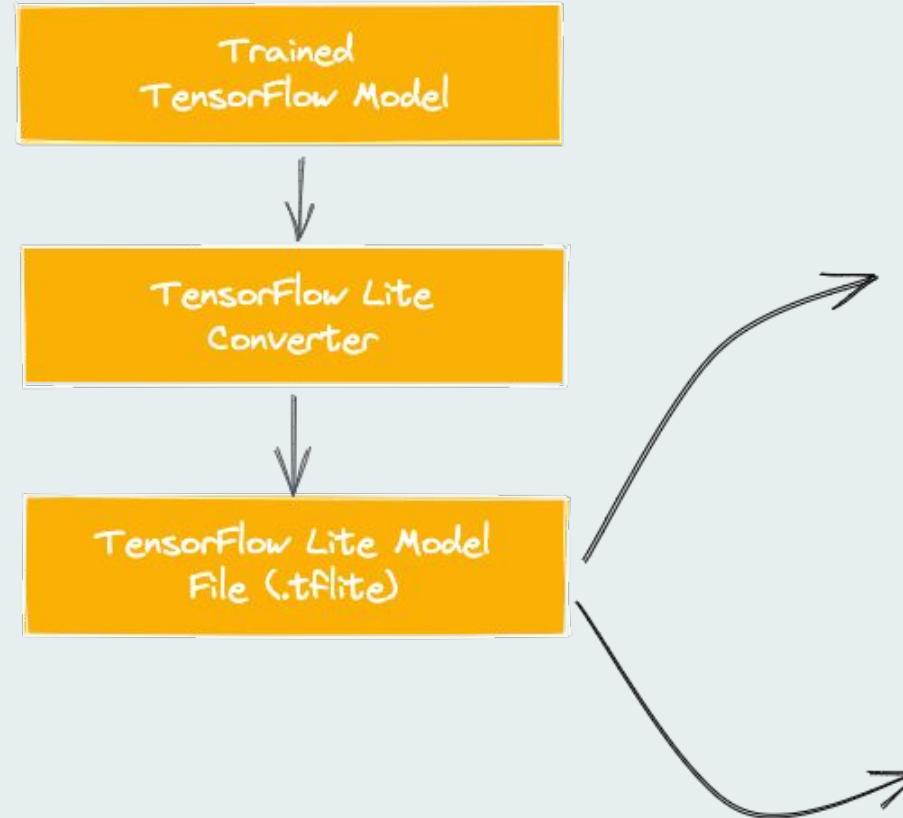
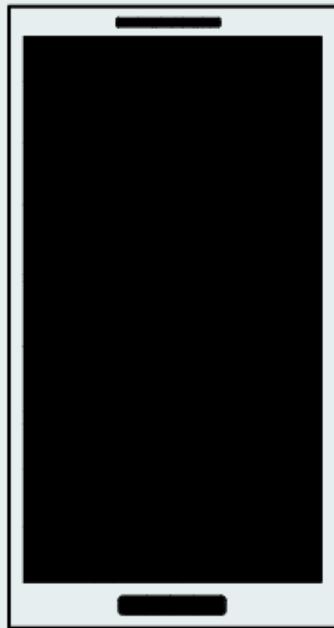


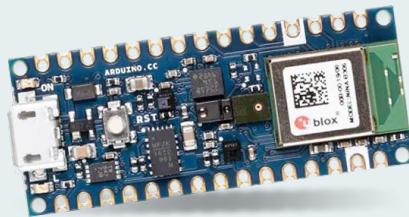
TensorFlow Lite

Key Differences

	 TensorFlow	 TensorFlow Lite
Topology	Variable	Fixed
Weights	Variable	Fixed
Binary Size	Unimportant	High Priority
Distributed Compute	Needed	Not Needed
Developer Background	ML Researcher	Application Developer

Architecture





Even less memory

Even less computer power

Also, only focused on inference

