Artificial Intelligence Chips: From Data Centers to Edge and IoT Computing

Version 1.0, 2018, Updated on: 7/30/2020

Abderazek Ben Abdallah

Adaptive Systems Laboratory

benab@u-aizu.ac.jp

AI Hardware is … everywhere

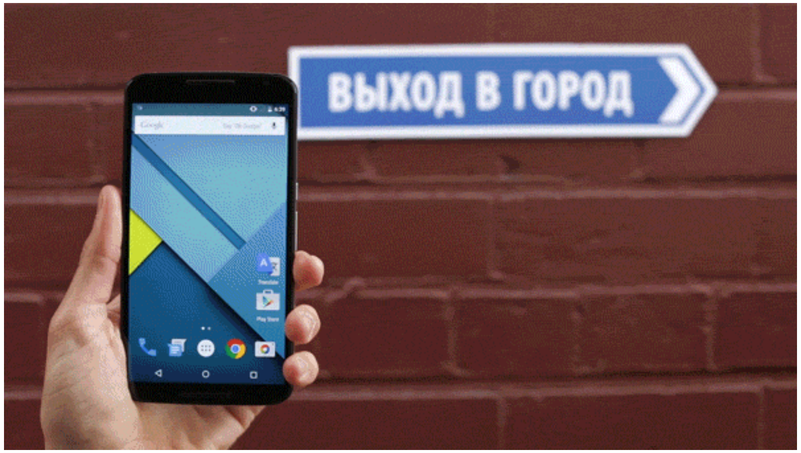
Self-driving Car

Bottom Image source: edition.cnn.com

Smart Robots

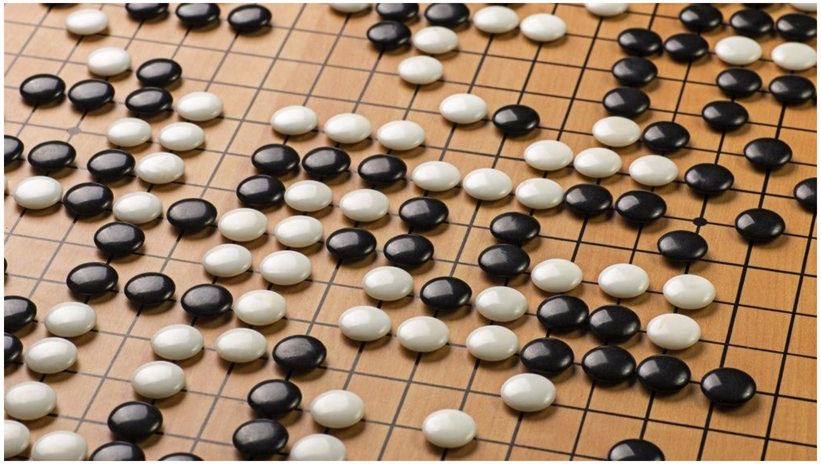
Image source: roboticsbusinessreview.com

Machine Translation



Bottom Image source: missqt.com

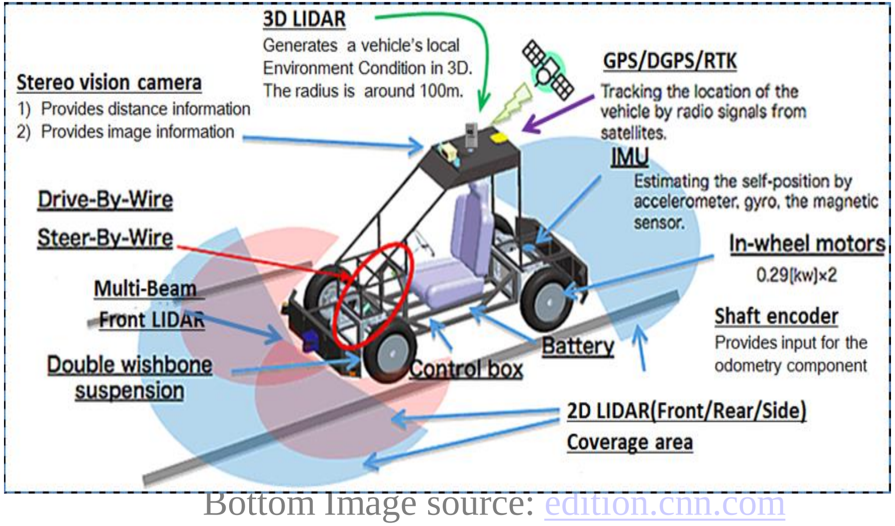
Gaming



Bottom Image Source: newatlas.com

AI Hardware is … everywhere

Self-driving Car

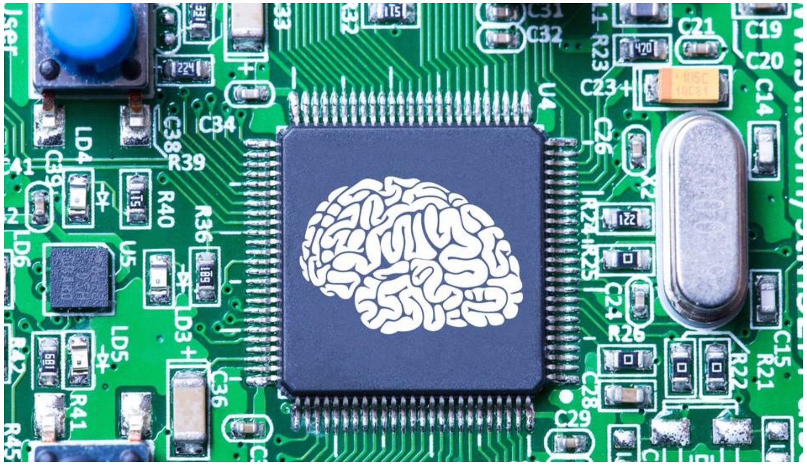

3D LIDAR
Generates a vehicle's local Environment Condition in 3D. The radius is around 100m.
Stereo vision camera
1) Provides distance information
2) Provides image information
GPS/DGPS/RTK
Tracking the location of the vehicle by radio signals from satellites.
IMU
Estimating the self-position by accelerometer, gyro, the magnetic sensor.
Drive-By-Wire
Steer-By-Wire
In-wheel motors
0.29[kw]×2
Multi-Beam
Front LIDAR
Shaft encoder
Provides input for the odometry component
Battery
Double wishbone suspension
Control box
2D LIDAR(Front/Rear/Side)
Coverage area

Bottom Image source: edition.cnn.com

Smart Robots

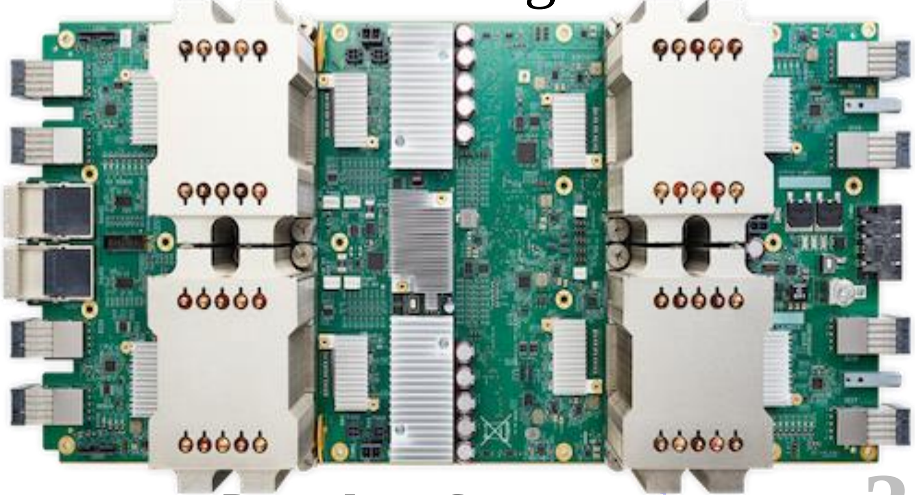
Image source: roboticsbusinessreview.com

Machine Translation



Bottom Image source: missqt.com

Gaming



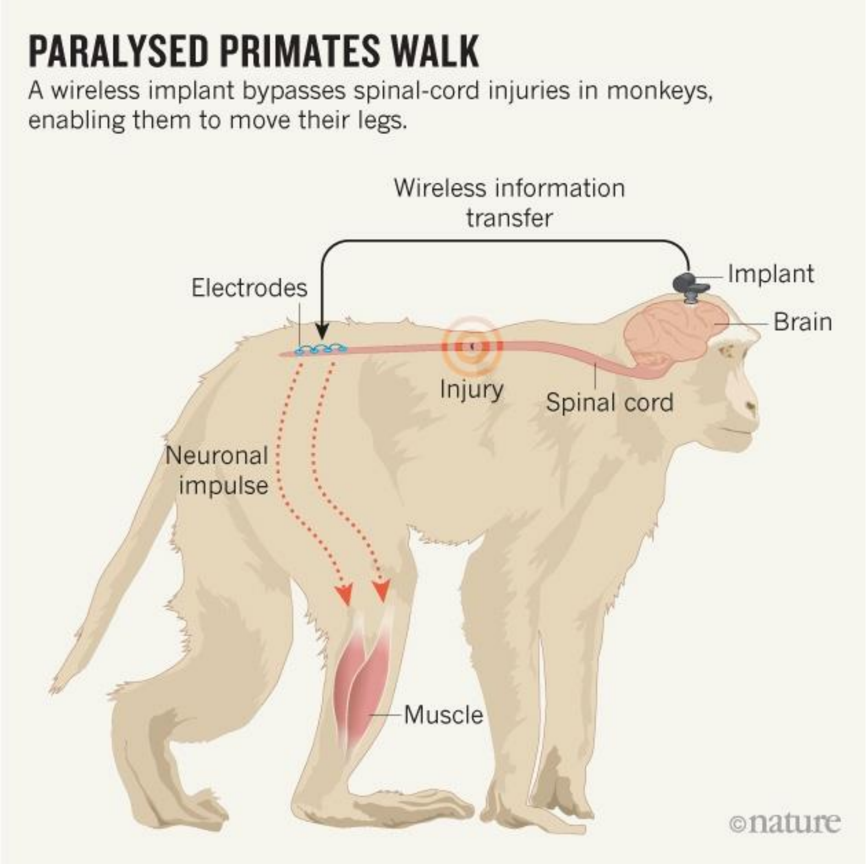
Bottom Image Source: newatlas.com

AI Hardware is … everywhere

Brain implant allows paralysed monkey to walk

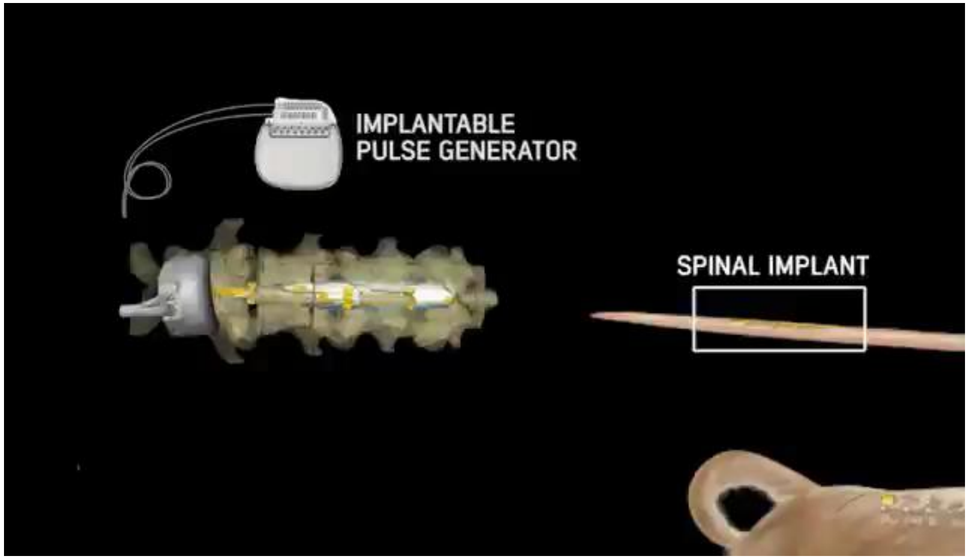
There really is a kind of intelligence inside the spinal cord. We are not just talking about reflexes that automatically activate muscles. In the spinal cord there are networks of neurons able to take their own decisions

-Grégoire Courtine-
Neuroscientist, Federal Institute of Technology, Lausanne



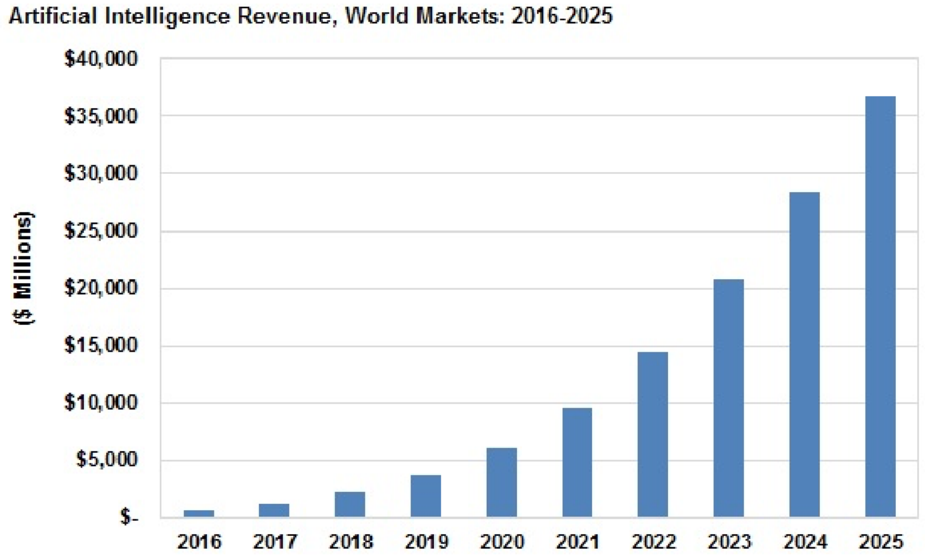
PARALYSED PRIMATES WALK

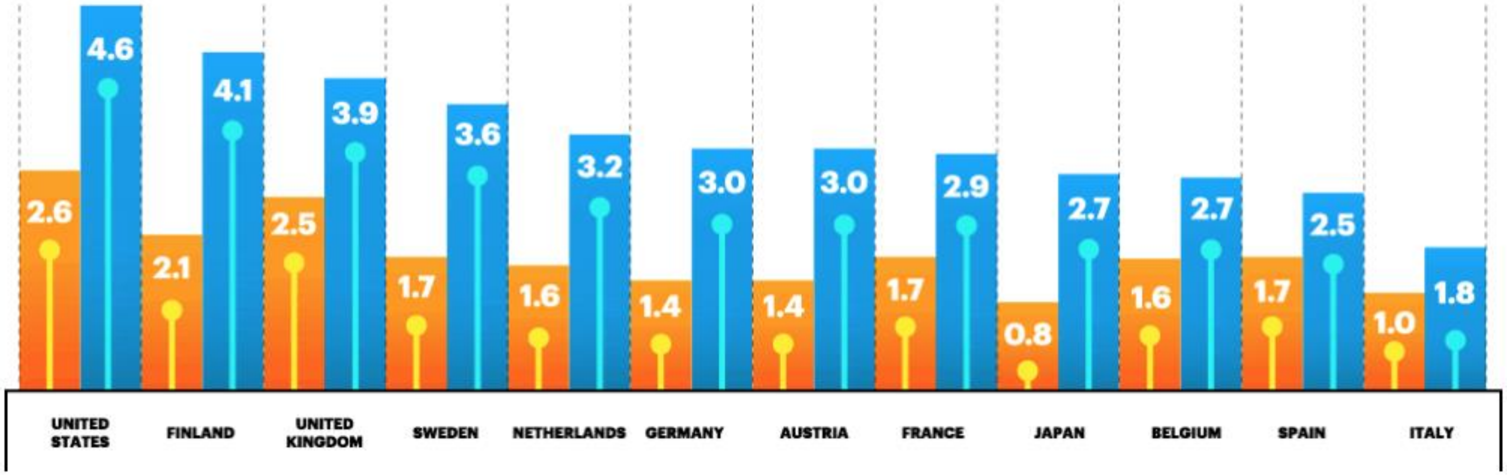
A wireless implant bypasses spinal-cord injuries in monkeys, enabling them to move their legs.



Nature volume539, pages284–288 (10 November 2016)

AI Revenue & GDP Growth Rate in 2035 comparing Baseline Growth to AI scenario





Annual growth rates in 2035 of gross value added (a close approximation of GDP), comparing baseline growth in 2035 to an artificial intelligence scenario where AI has been absorbed into the economy

Source: Accenture and Frontier Economics

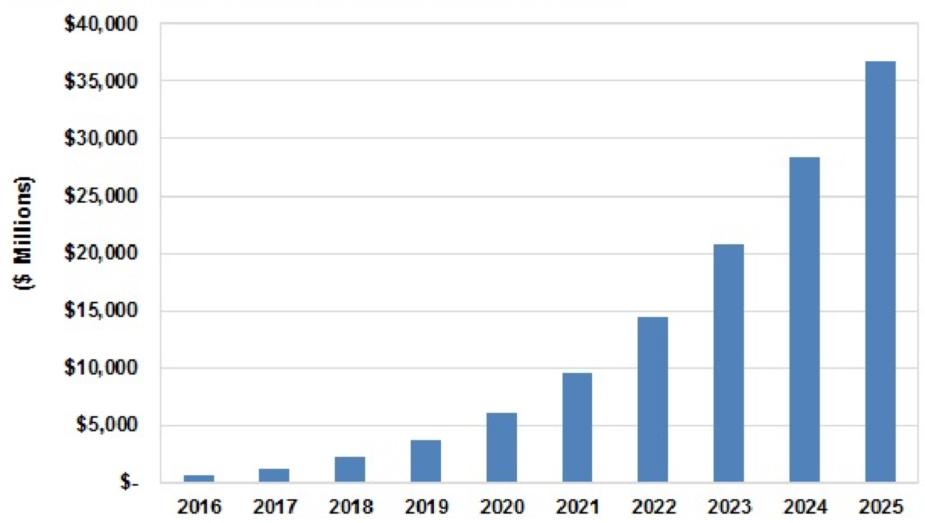
🟧 Baseline

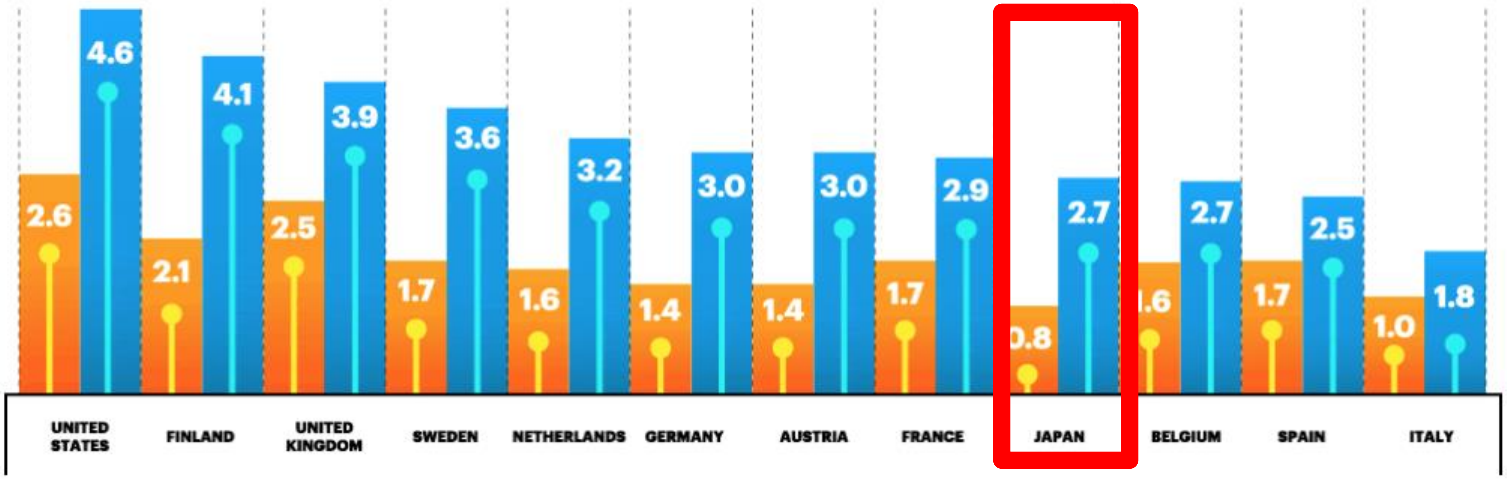
🟦 AI steady rate

Source: https://semiengineering.com/what-does-an-ai-chip-look-like/

AI Revenue &
GDP Growth Rate in 2035 comparing Baseline Growth to AI scenario

Artificial Intelligence Revenue, World Markets: 2016-2025





Annual growth rates in 2035 of gross value added (a close approximation of GDP), comparing baseline growth in 2035 to an artificial intelligence scenario where AI has been absorbed into the economy

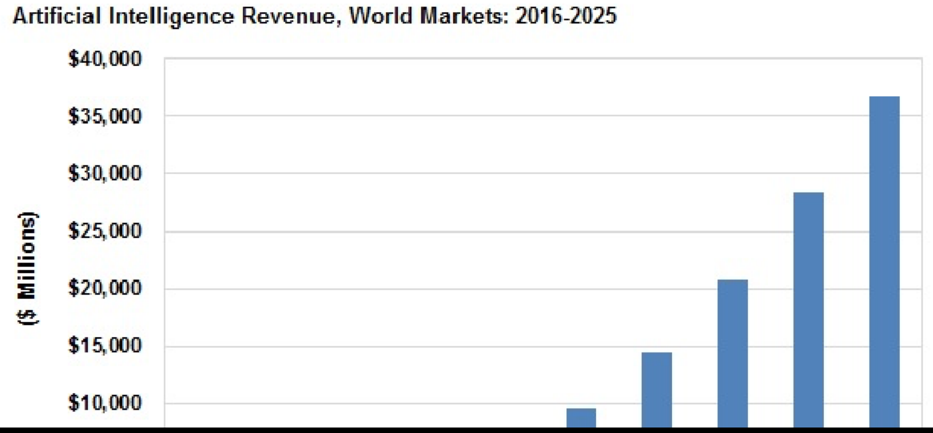
Source: Accenture and Frontier Economics

🟧 Baseline

🟦 AI steady rate

Source: https://semiengineering.com/what-does-an-ai-chip-look-like/

AI Revenue & GDP Growth Rate in 2035 comparing Baseline Growth to AI scenario



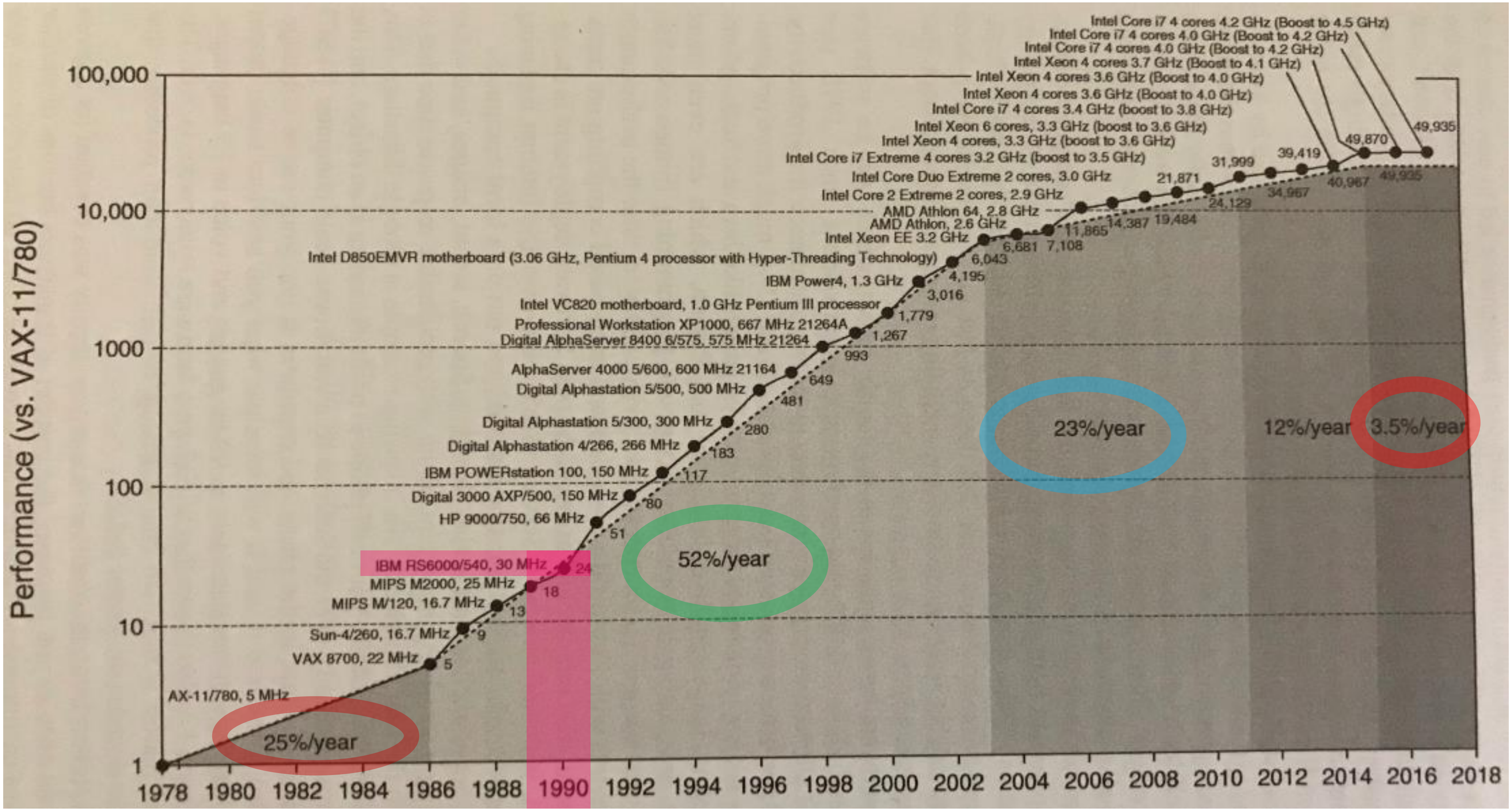
Governments are competing to establish advanced AI research, seeing AI as a way for greater economic power and influence.

Source: CBINSPIGHTS 2018

Agenda

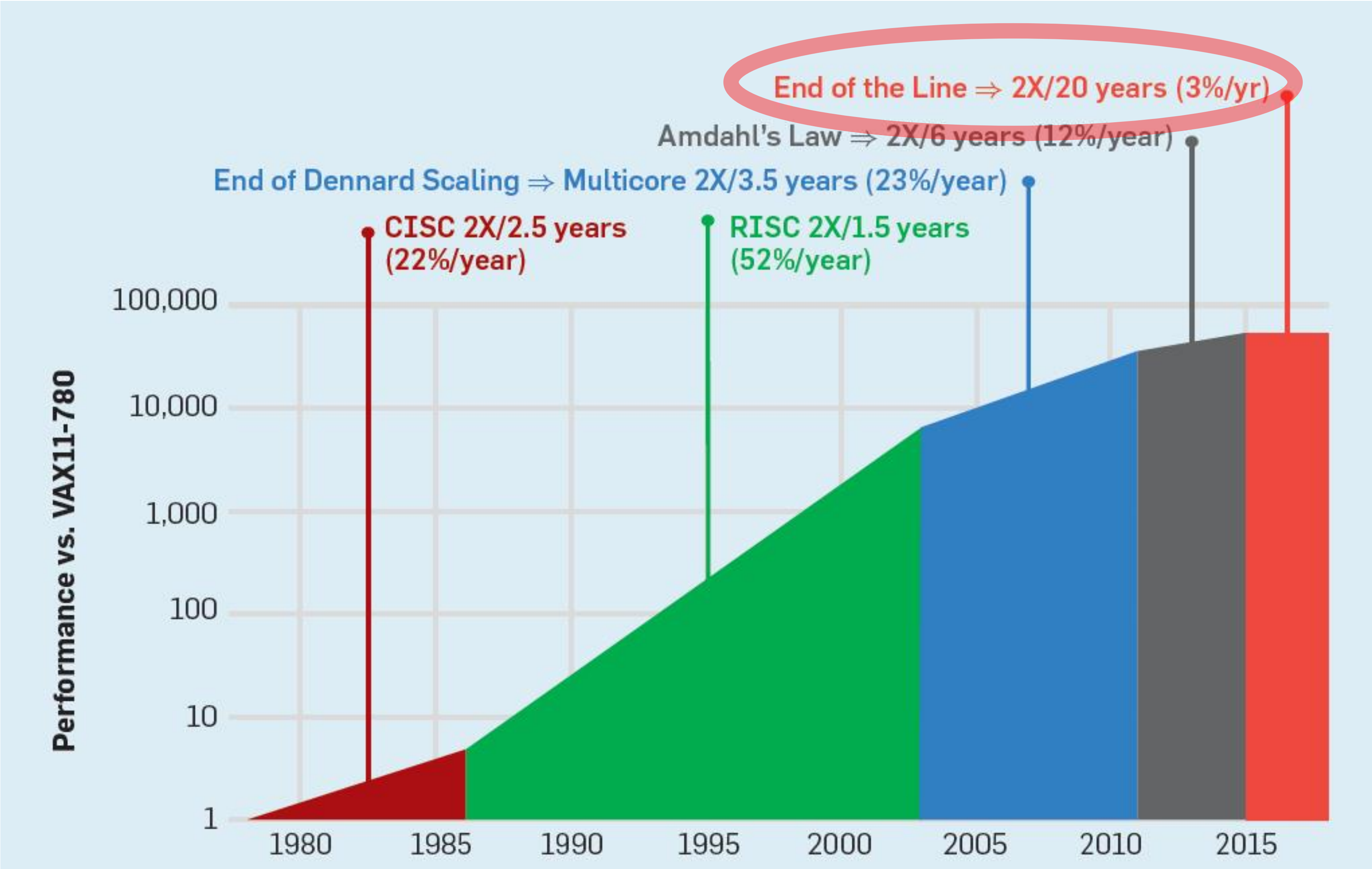
- Fundamental Trends
- **AI – The 4th Industrial Revolution**
- **Survey of AI Hardware**
  - Cloud AI Hardware, Chips
  - Mobile AI Chips
  - Edge and IoT AI Chips
  - Healthcare AI Chips
- **Conclusions**

Moore’s law is no longer providing more Compute



Source: Norman P. Jouppi, Cliff Young, Nishant Patil, David Patterson, Communications of the ACM, September 2018, Vol. 61 No. 9, Pages 50-59

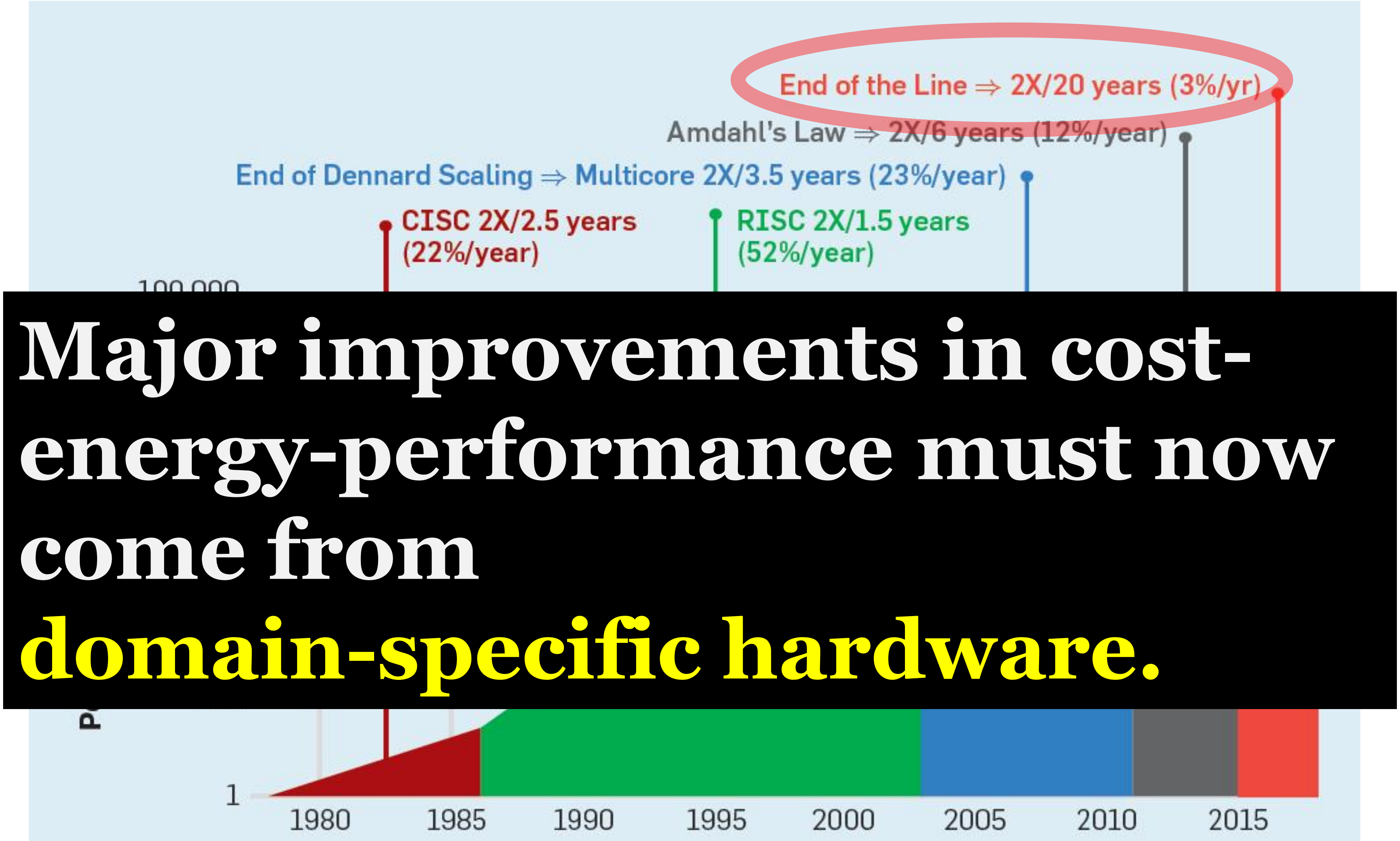
Moore’s law is no longer providing more compute



****Dennard scaling**: As transistors get smaller their power density stays constant, so that the power consumption stays in proportion with area: both voltage and current scale (downward) with length (WP).

Source: Norman P. Jouppi, Cliff Young, Nishant Patil, David Patterson, Communications of the ACM, September 2018, Vol. 61 No. 9, Pages 50-59

Moore's law is no longer providing more compute



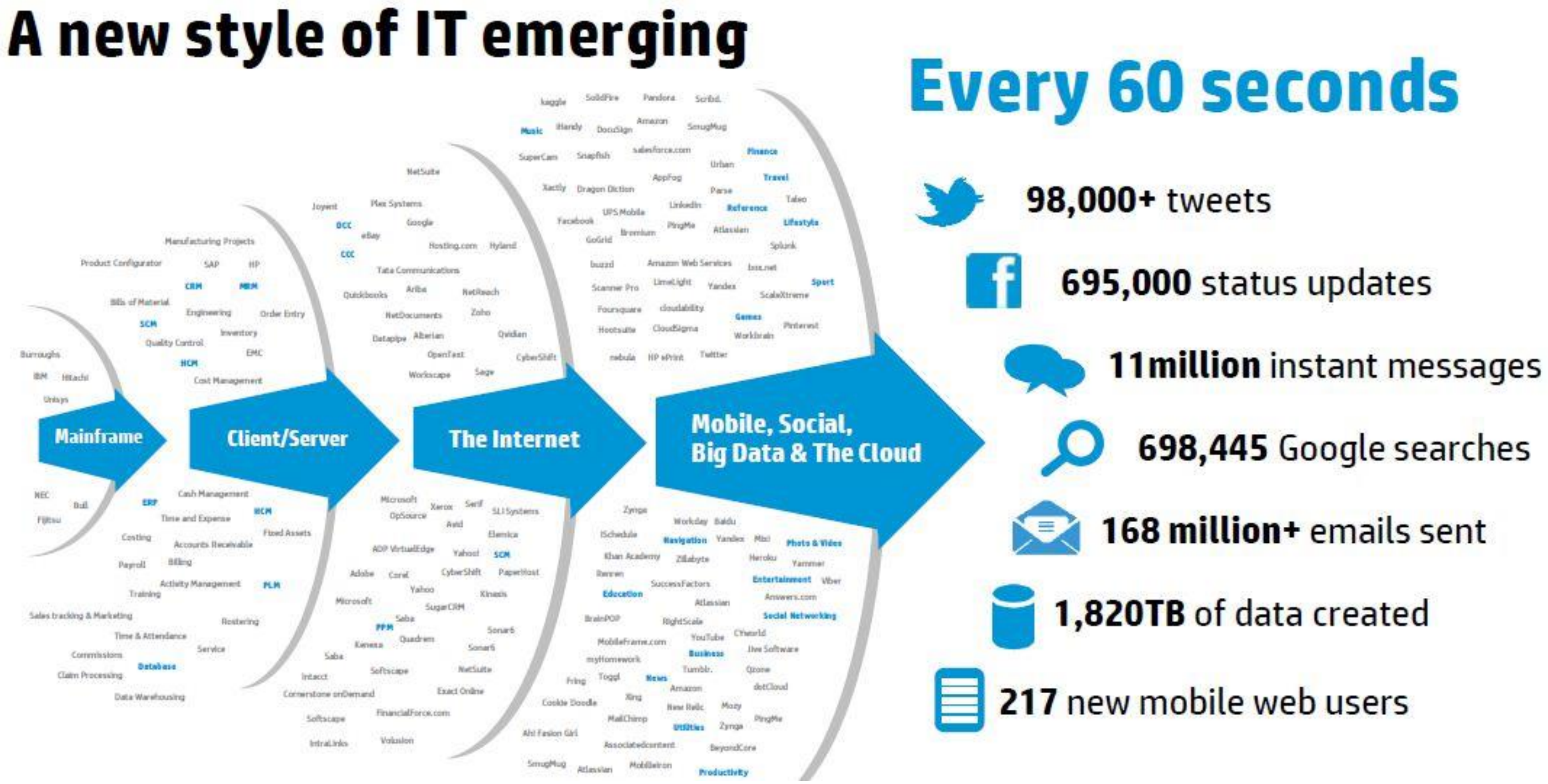
Major improvements in cost-energy-performance must now come from domain-specific hardware.

****Dennard scaling**: As transistors get smaller their power density stays constant, so that the power consumption stays in proportion with area: both voltage and current scale (downward) with length (WP).

Source: Norman P. Jouppi, Cliff Young, Nishant Patil, David Patterson, Communications of the ACM, September 2018, Vol. 61 No. 9, Pages 50-59

Technology Transformation

Massive amounts of data is generated


A new style of IT emerging
Mainframe
Client/Server
The Internet
Mobile, Social, Big Data & The Cloud

Every 60 seconds

- 98,000+ tweets
- 695,000 status updates
- 11million instant messages
- 698,445 Google searches
- 168 million+ emails sent
- 1,820TB of data created
- 217 new mobile web users

Source: https://practicalanalytics.files.wordpress.com/2012/10/newstyleofit.jpg

DNN Compute Requirements is Steadily Growing

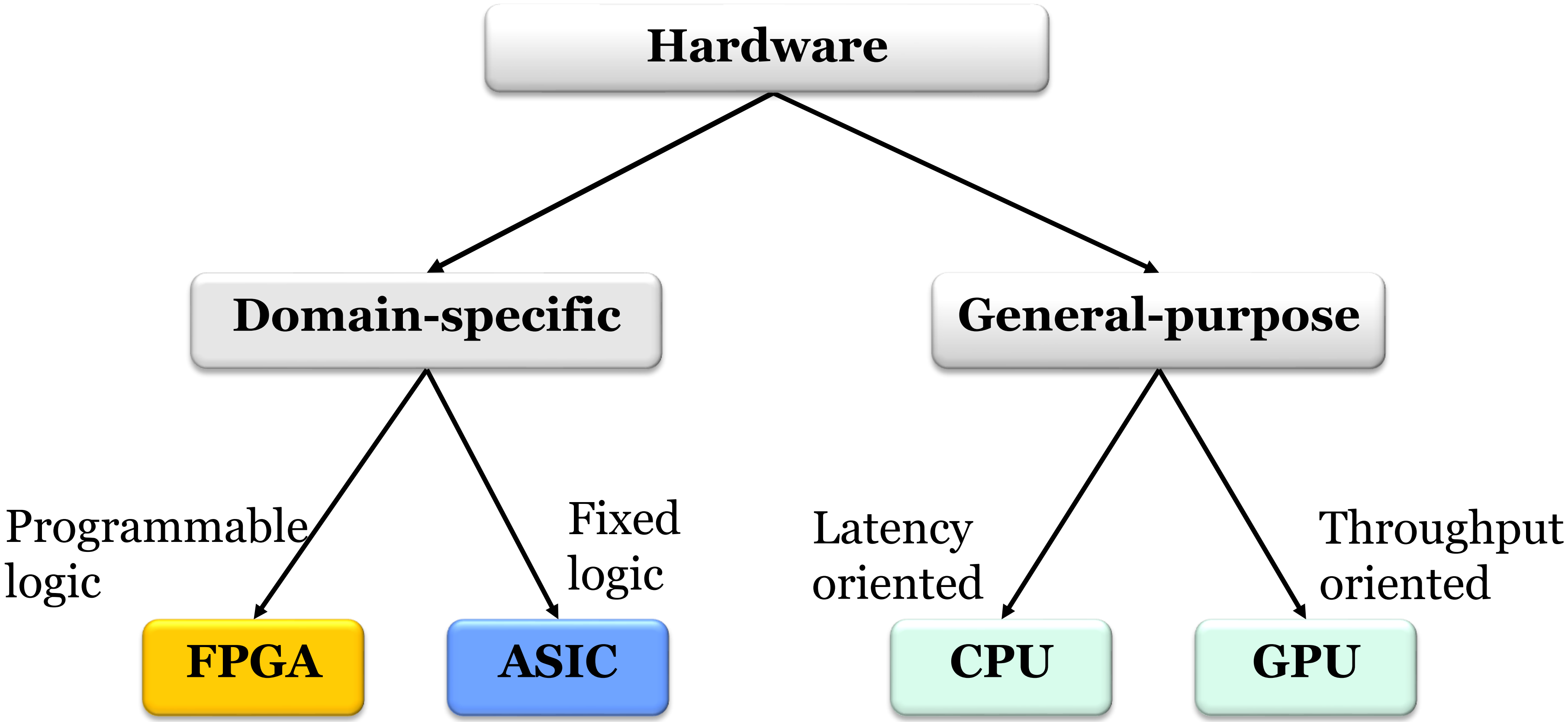
Metrics	LeNet-5	AlexNet	VGG-16	GoogLeNet (v1)	ResNet-50
Top-5 error	n/a	16.4	7.4	6.7	5.3
Input Size	28x28	227x227	224x224	224x224	224x224
# of CONV Layers	**2**	**5**	**16**	**21 (depth)**	**49**
Filter Sizes	5	3, 5,11	3	1, 3 , 5, 7	1, 3, 7
# of Channels	1, 6	3 - 256	3 - 512	3 - 1024	3 - 2048
# of Filters	6, 16	96 - 384	64 - 512	64 - 384	64 - 2048
Stride	1	1, 4	1	1, 2	1, 2
# of Weights	2.6k	2.3M	14.7M	6.0M	23.5M
# of MACs	283k	666M	15.3G	1.43G	3.86G
# of FC layers	**2**	**3**	**3**	**1**	**1**
# of Weights	58k	58.6M	124M	1M	2M
# of MACs	58k	58.6M	124M	1M	2M
Total Weights	**60k**	**61M**	**138M**	**7M**	**25.5M**
Total MACs	**341k**	**724M**	**15.5G**	**1.43G**	**3.9G**

Source: Joel Emer, ISCA Tutorial, 2017

What does it mean ?

End of Moore’s Law + Exponential Increase in Compute Requirements = Needs New Approach

Current State of the Art in Neural Algorithms HW Computing



Hardware

Domain-specific

General-purpose

Programmable logic

Fixed logic

Latency oriented

Throughput oriented

FPGA

ASIC

CPU

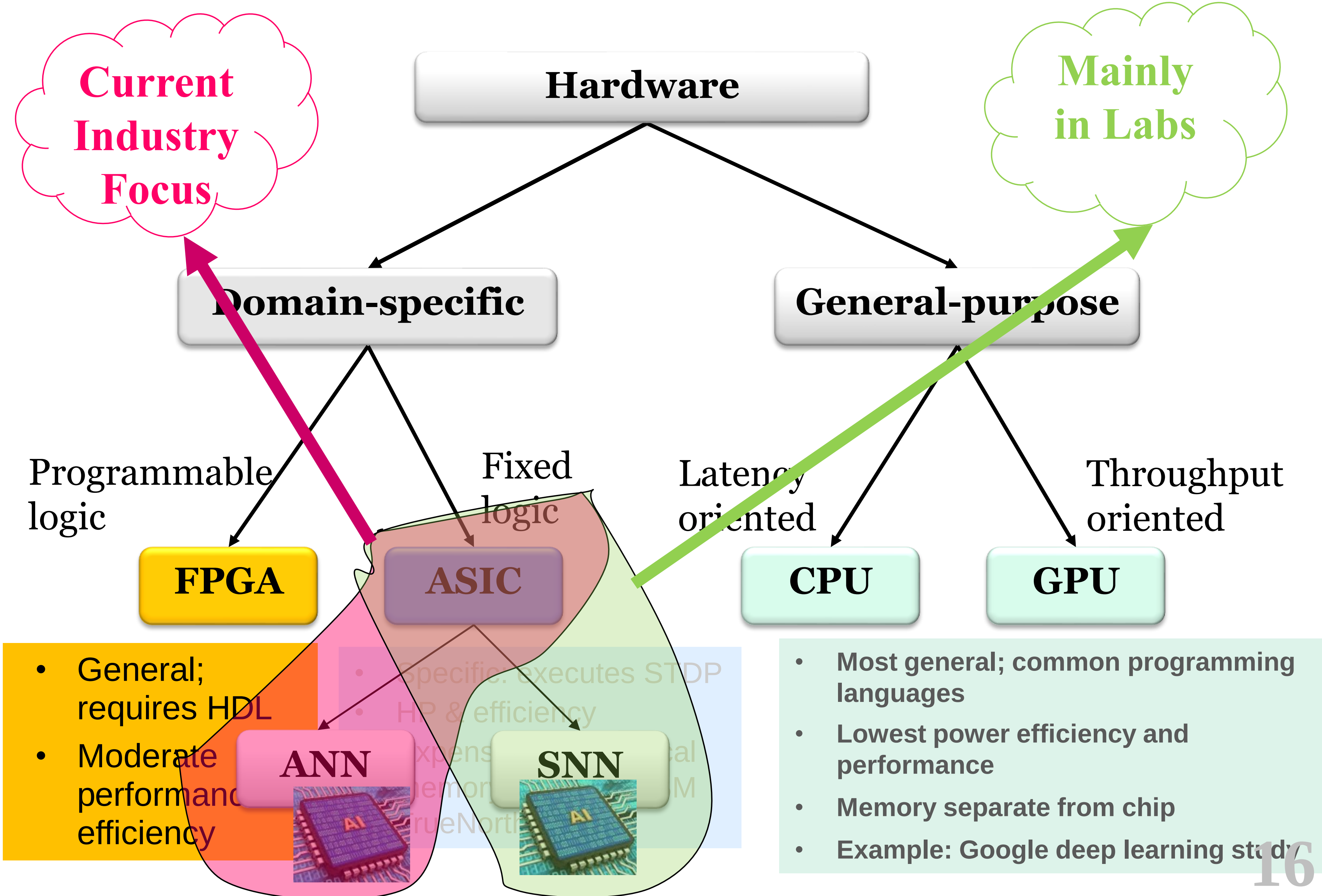
GPU

- General; requires HDL
- Moderate performance & efficiency

- Specific: executes STDP
- HP & efficiency
- Expensive, 40MB local memory Example: IBM TrueNorth

- Most general; common programming languages
- Lowest power efficiency and performance
- Memory separate from chip
- Example: Google deep learning study

Current State of the Art in Neural Algorithms HW Computing



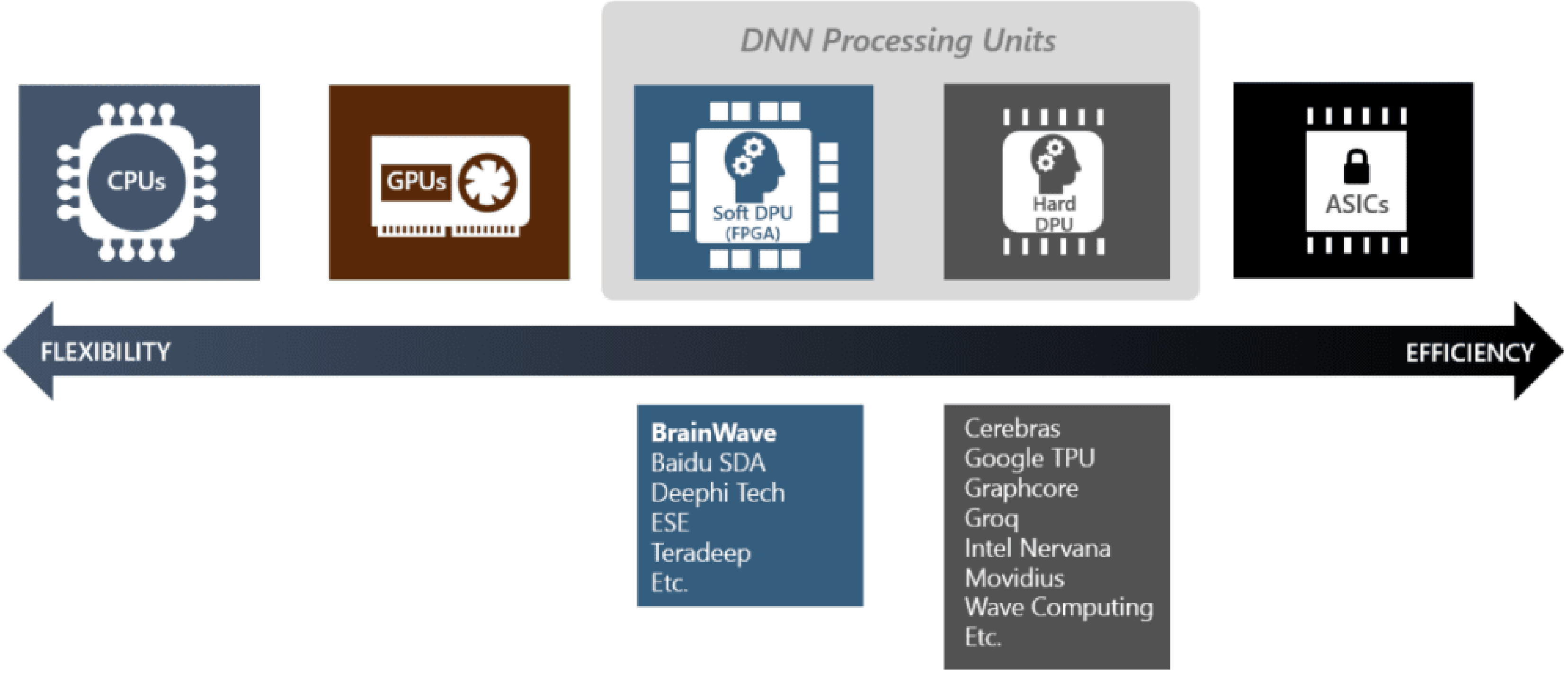
- General; requires HDL
- Moderate performance efficiency

- Specific; executes STDP
- HP & efficiency
- Expens[illegible]cal
- memory [illegible]M
- TrueNorth

- Most general; common programming languages
- Lowest power efficiency and performance
- Memory separate from chip
- Example: Google deep learning study

CPUs, GPUs, FPGAs or ASICs ?

The only tricky part is getting them to do AI computation quickly and efficiently.



BrainWave
Baidu SDA
Deephi Tech
ESE
Teradeep
Etc.

Cerebras
Google TPU
Graphcore
Groq
Intel Nervana
Movidius
Wave Computing
Etc.

Hardware: Flexibility vs Efficiency

Deployment alternatives for deep neural networks (DNNs) and examples of their implementations. (Image courtesy of Microsoft.)

Agenda

- **Fundamental Trends**
- **AI – The 4th Industrial Revolution**
- **Survey of AI Hardware**
  - ➢ Cloud AI Hardware, Chips
  - ➢ Mobile AI Chips
  - ➢ Edge and IoT AI Chips
  - ➢ Healthcare AI Chips
- **Conclusions**

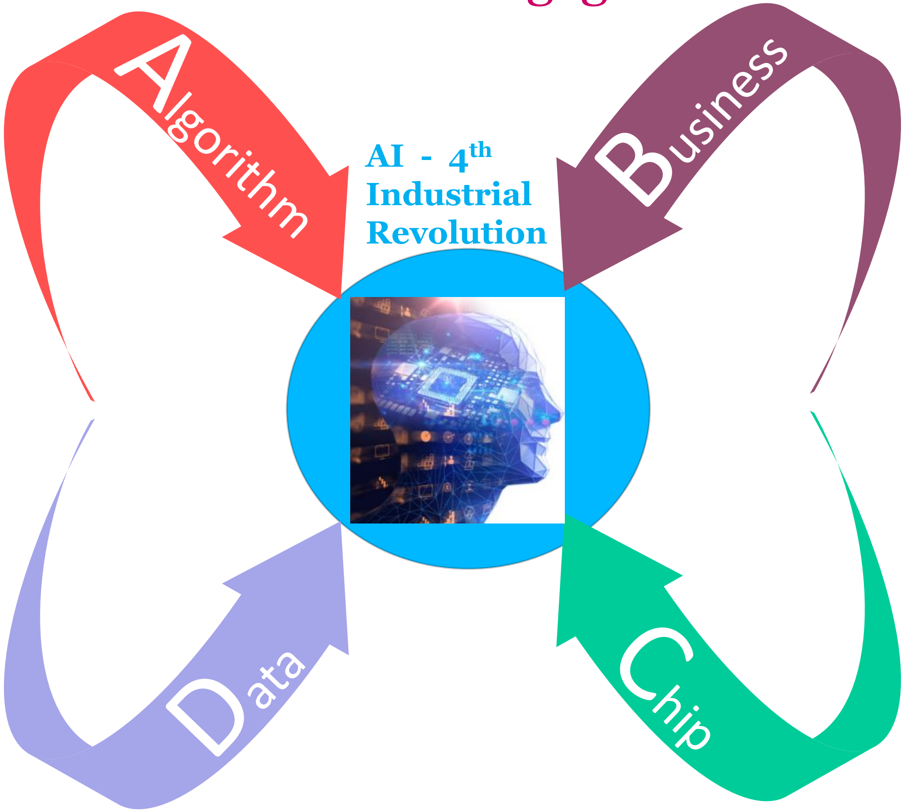
Four Main Factors in Promoting AI/AI HW

Image:kdnuggets.com

AI algorithms are being applied to nearly everything we do.

Strong Gov. & Industry Engagements

Image: kdnuggets.com



Growth of computational power

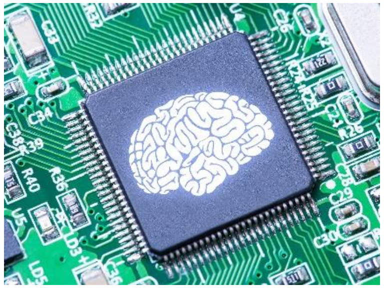


Image: spectrum.ieee.org

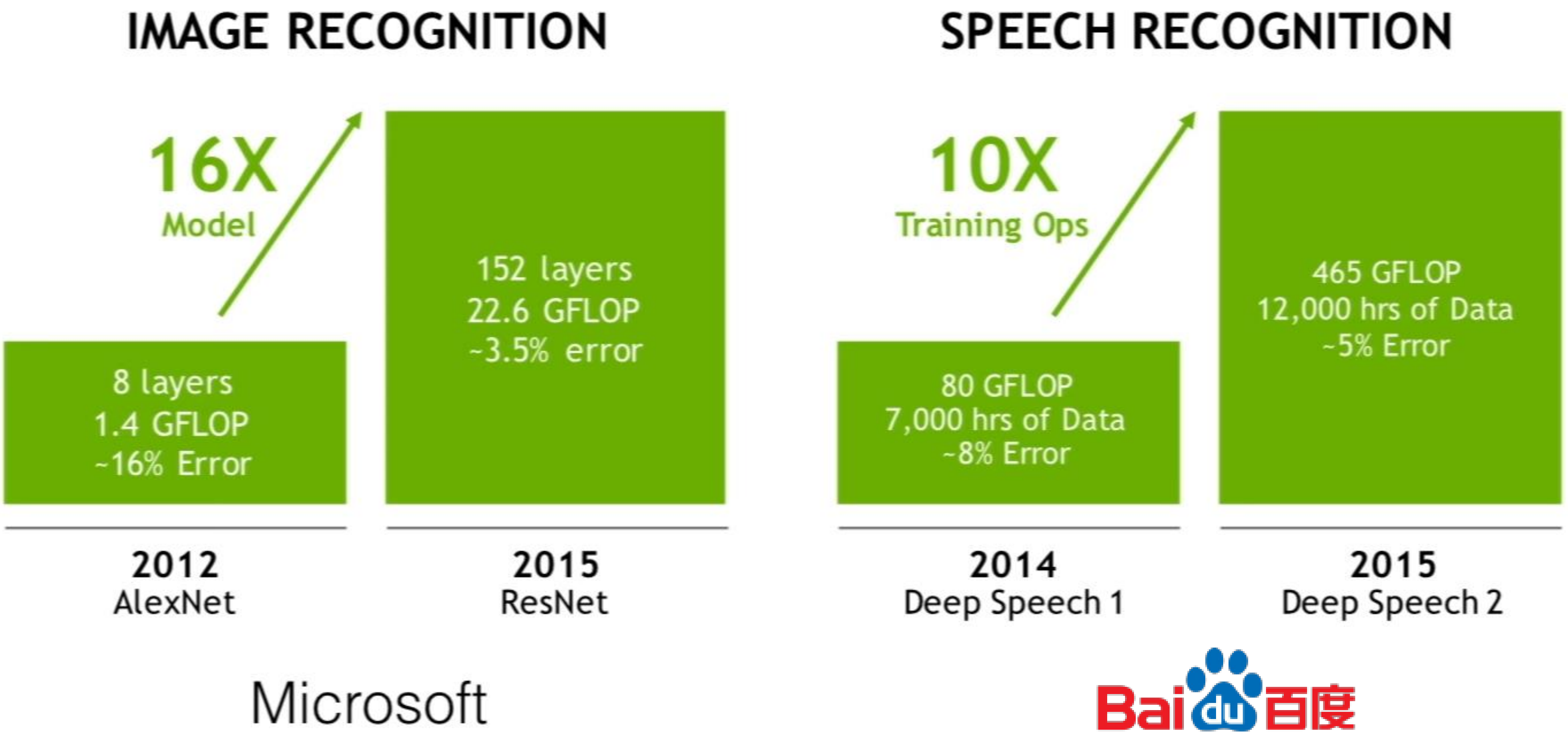
Image: sas.com

Larger data sets and models lead to better accuracy but also increase the computation time

More compute means new solutions to previously intractable problems, i.e. G

Hardware & Data Enable DNNs

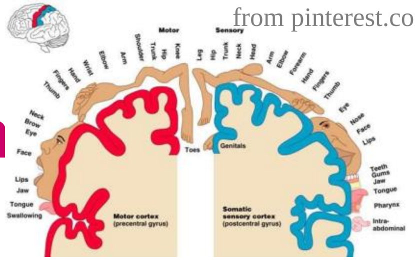
AI model performance scales with dataset size and the # of model parameters, thus necessitating more compute.



Dally, NIPS'2016 workshop on Efficient Methods for Deep Neural Networks

AI HW is inspired by Nature – Biological neuron

AI Chips and systems are inspired by biology ➔ parallel computation



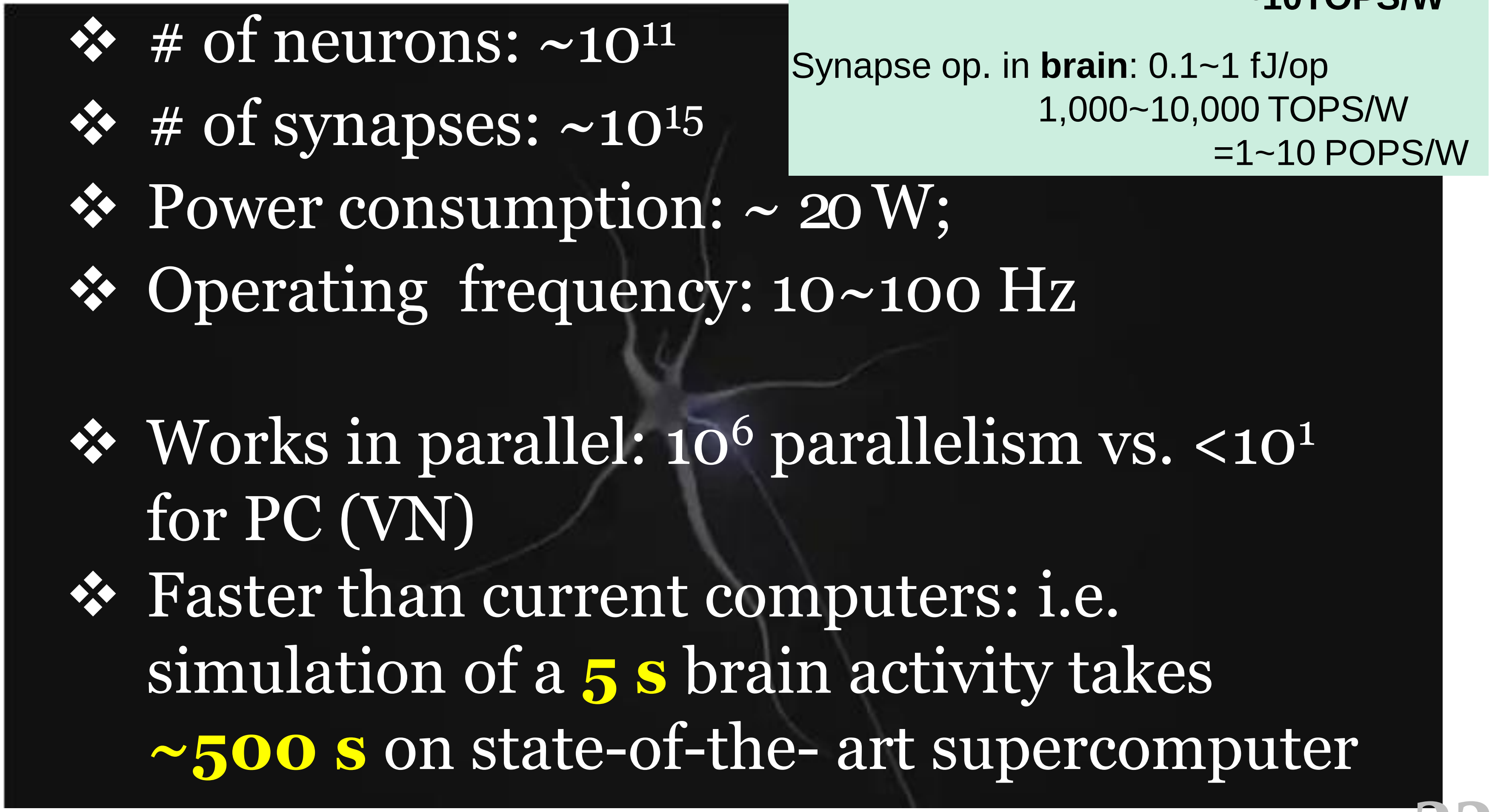
from pinterest.com

AI HW is inspired by Nature – Biological neuron

AI Chips and systems are inspired by biology ➔ parallel computation.

Latest digital DL processors:
~10TOPS/W

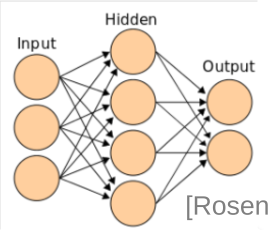
Synapse op. in **brain**: 0.1~1 fJ/op
1,000~10,000 TOPS/W
=1~10 POPS/W



- # of neurons: ~10^{11}
- # of synapses: ~10^{15}
- Power consumption: ~ 20 W;
- Operating frequency: 10~100 Hz

- Works in parallel: 10^{6} parallelism vs. <10^{1} for PC (VN)
- Faster than current computers: i.e. simulation of a **5 s** brain activity takes **~500 s** on state-of-the- art supercomputer

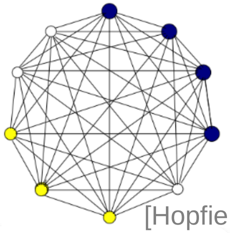
…there are many topologies for mimicking the brain functions



[Rosenblatt,1958]

1954

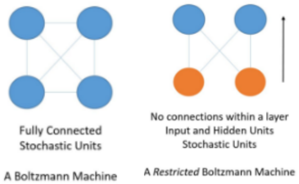
- F-Forward (FFNN)



[Hopfield,1982]

1982

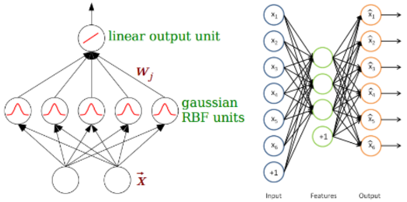
- Hopfield (HN)


Fully Connected Stochastic Units
A Boltzmann Machine
No connections within a layer Input and Hidden Units Stochastic Units
A Restricted Boltzmann Machine

Hinton,1986]

1986

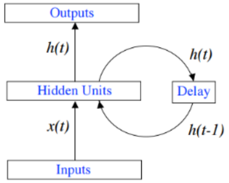
- **Boltzmann, Restricted Boltzmann (RBM)**



[Broomhead,1988]

1988

- Radial Bas. Func, Autoencoder (AE)



[Elman,1990]

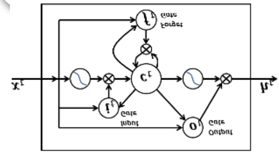
1990

- Recurrent Neural (RNN)

[Hochreiter,1997]

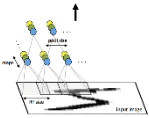
1997

- Long/Short Term Mem (LSTM), BiRNN



1998

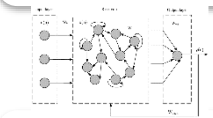
- Conv. Neural Network (CNN)



[LeCum, 1998]

2004

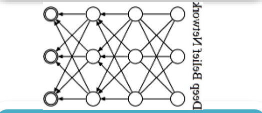
- Echo State Network (ESN)



[Jaeger, 2004]

2007

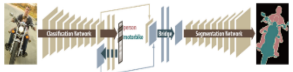
- **Deep Belief Net (DBN), SAE**



[Bengio, 2007]

2010

- **Deconvolutional Network (DN)**

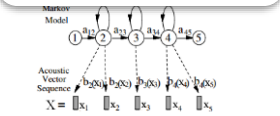


[Zeiler, 2010]

[Hayes, 2013]

2013

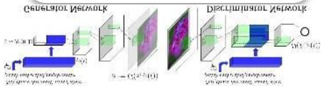
- **Markov Chain (MC), VAE, ELM**



Ex.
I: red flower
O: red flower image

2014

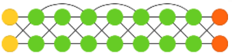
- Generartive adversarial net(GAN, GRU, NTM



[Goodfellow, 2015]

2015

- Deep Residual Net (DRN), DCIGN,

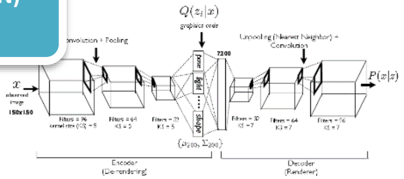


[He, 2015]

2015

- **Deep Convol. Inverse graphics Net (DCIGN)**

[Kulkarni, 2015]



Different approaches to AI Chips

Poor/Simple

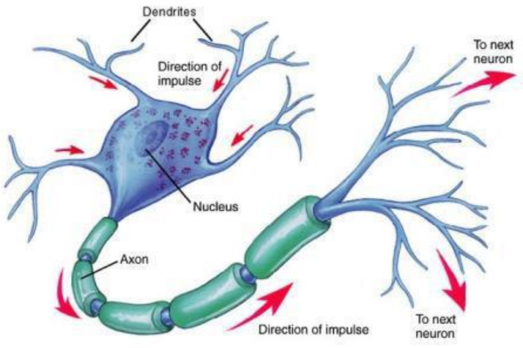
Good/Complex

Neuron Digital, Analog. LIF. . . . Izhikevich model Huxley-Hodgkin model . . .

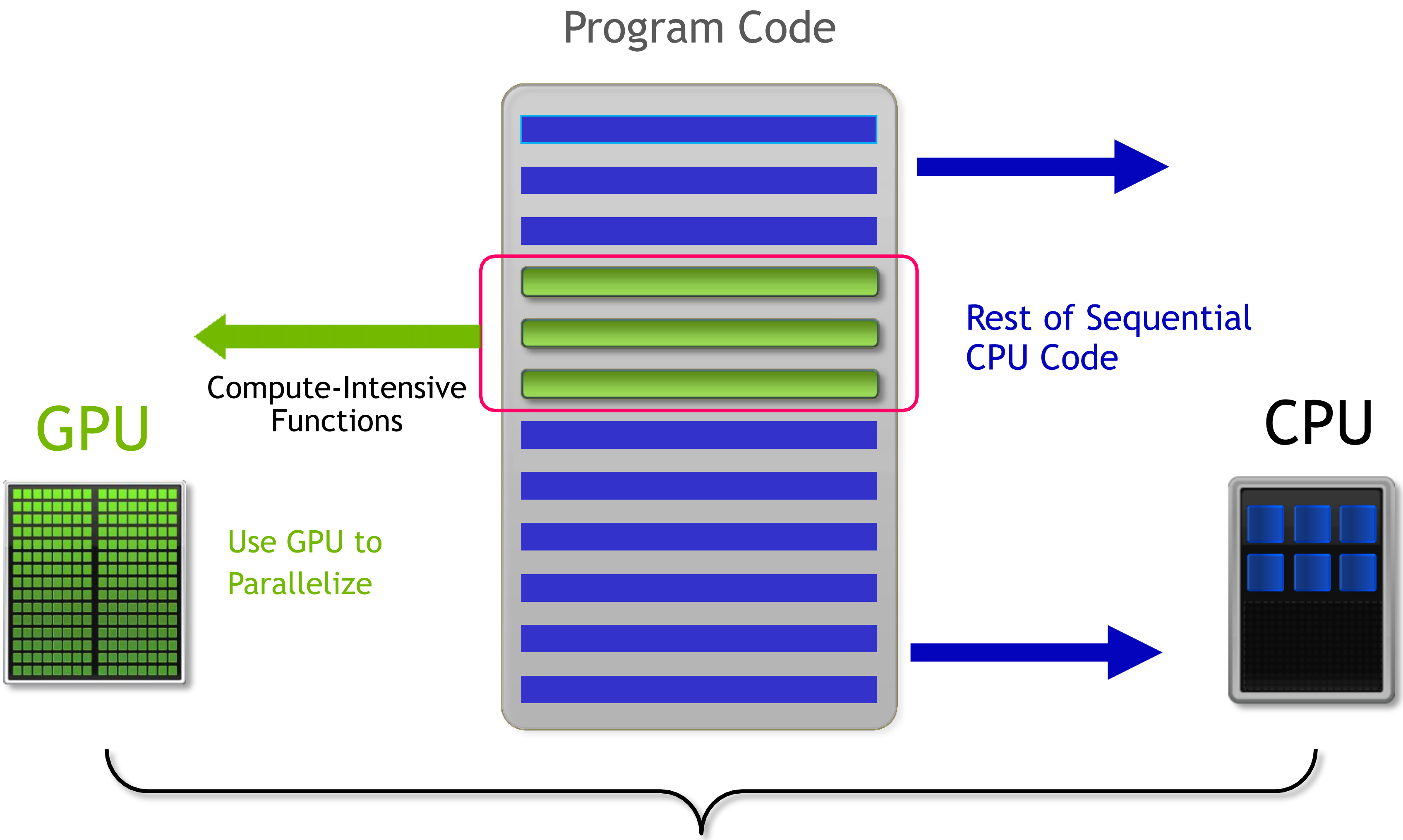
Synapse MAC (weighted . . . sum) Spiking STDP . . . . Many nonlinear properties . . . . .

Generally Used in DL algorithms

Frequency 10~100 Hz (brain)

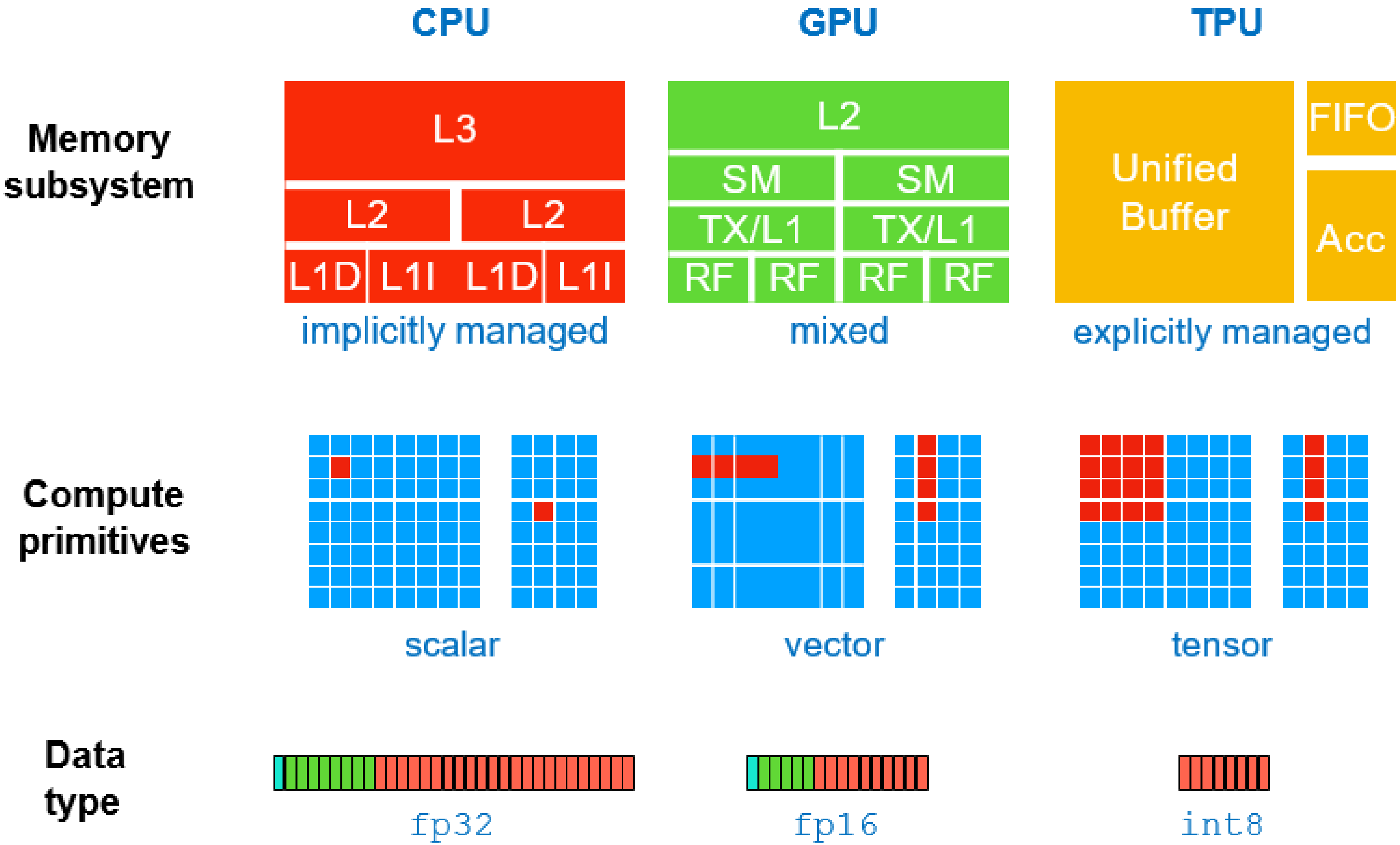


Current AI Chip = Accelerator/Co-processor



Acceleration with GPU

Accelerator Characteristics



CPU

GPU

TPU

Memory subsystem

L3

L2

L2

L1D

L1I

L1D

L1I

implicitly managed

L2

SM

SM

TX/L1

TX/L1

RF

RF

RF

RF

mixed

Unified Buffer

FIFO

Acc

explicitly managed

Compute primitives

scalar

vector

tensor

Data type

fp32

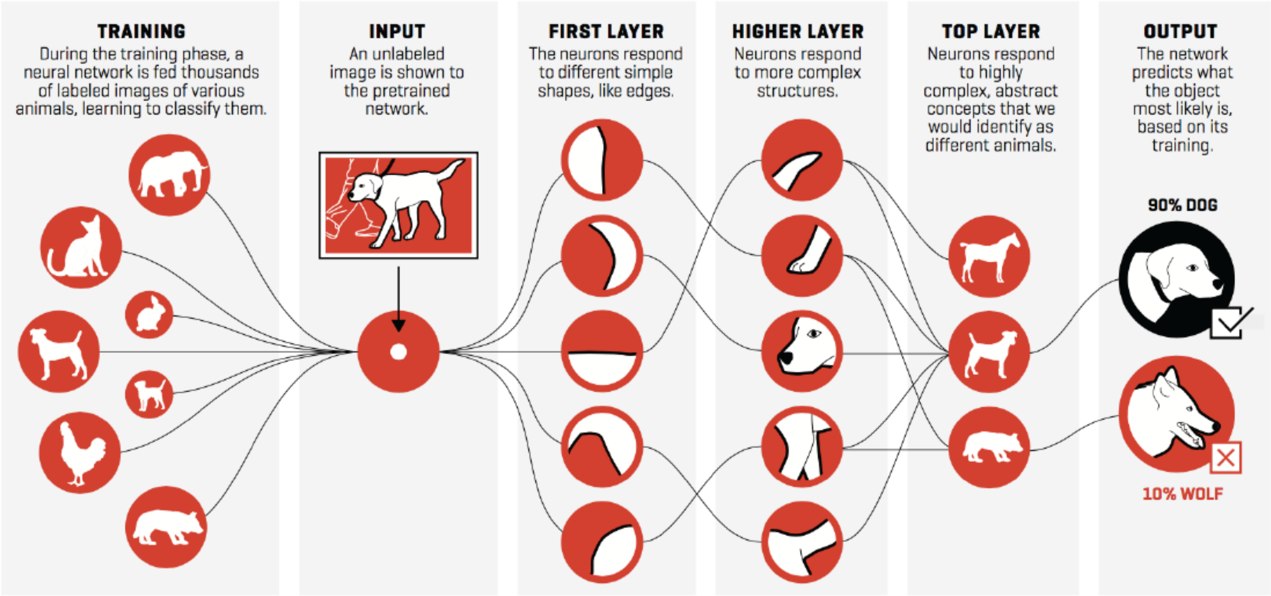
fp16

int8

[Ref 3]

…Deep Leering is considered as a sophisticated ''rocket'' of Machine Learning!!

Fuel = Data!



TRAINING
During the training phase, a neural network is fed thousands of labeled images of various animals, learning to classify them.

INPUT
An unlabeled image is shown to the pretrained network.

FIRST LAYER
The neurons respond to different simple shapes, like edges.

HIGHER LAYER
Neurons respond to more complex structures.

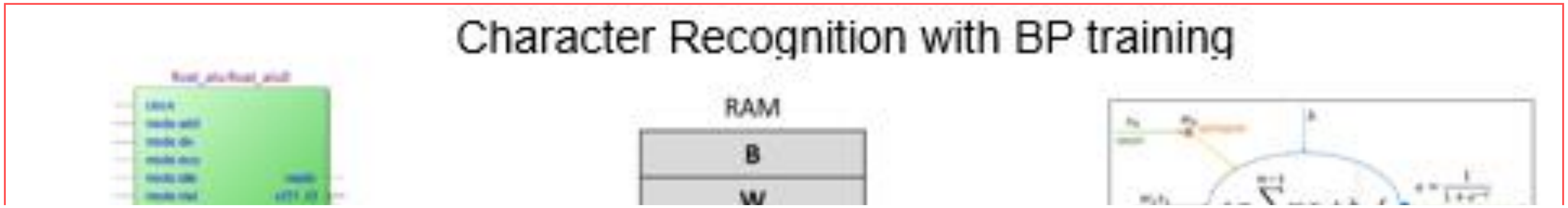
TOP LAYER
Neurons respond to highly complex, abstract concepts that we would identify as different animals.

OUTPUT
The network predicts what the object most likely is, based on its training.

1. ''Deep Learning'' means using a neural network with several layers of nodes between input & output

2. the series of layers between input & output do feature identification and processing in a series of stages, just as our brains seem to.

Example1: Character Recognition on FPGA



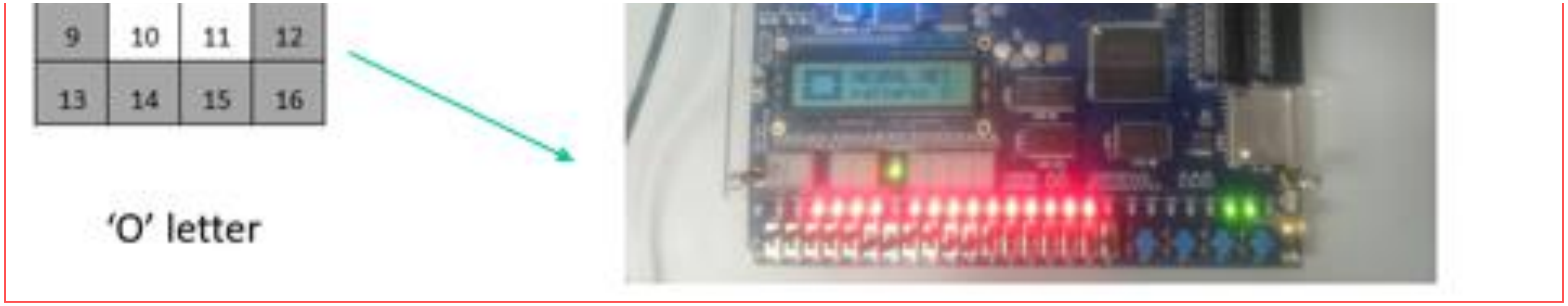
Implementation of detecting 16 patterns from 16 inputs with BP.

Device: EP2C35F672C6
Family: Cyclone2
Synthesis: Quartus2 13.1

Table 1 : ANN Performance Evaluation

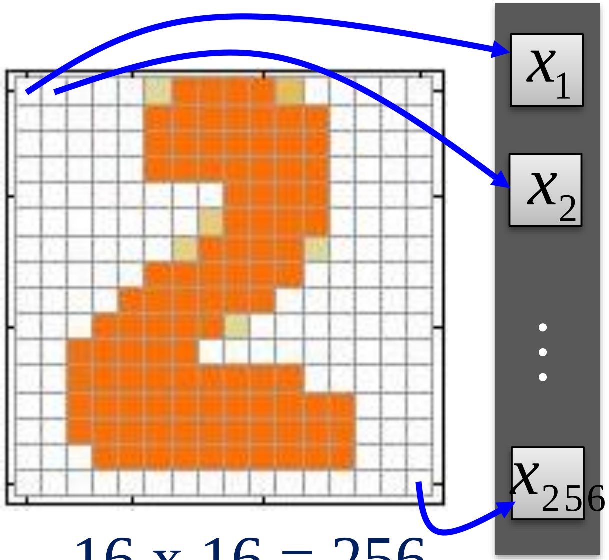
ALUs	Registers	Pins	Fmax
10,989 (33%)	5,814 (18%)	432 (89%)	76.02 MHz

Memory	DSP Block	Power Consumption	
4,956 (1%)	54 (77%)	286.84 mW	



Example2: Handwriting Digit Recognition on FPGA

Input



x_1

x_2

⋮

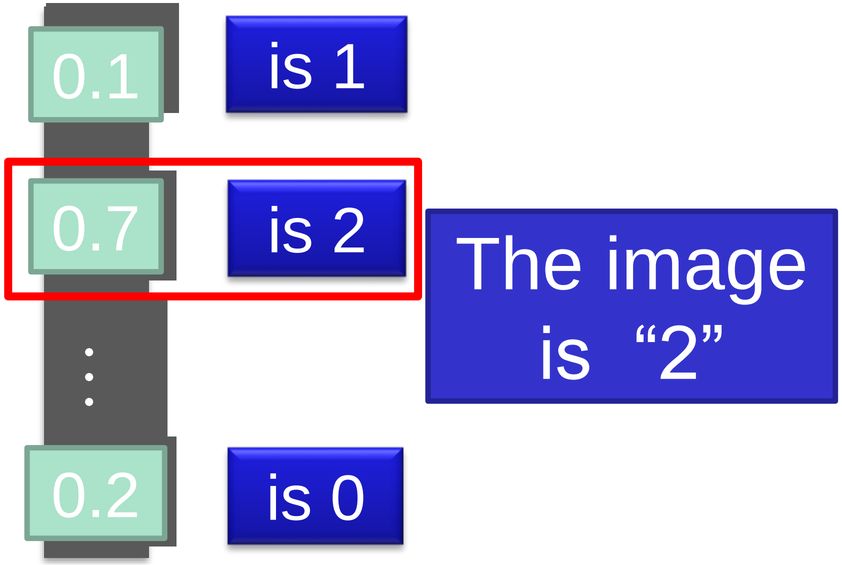
x_{256}

16 x 16 = 256

Ink → 1

No ink → 0

Output



0.1 is 1

0.7 is 2

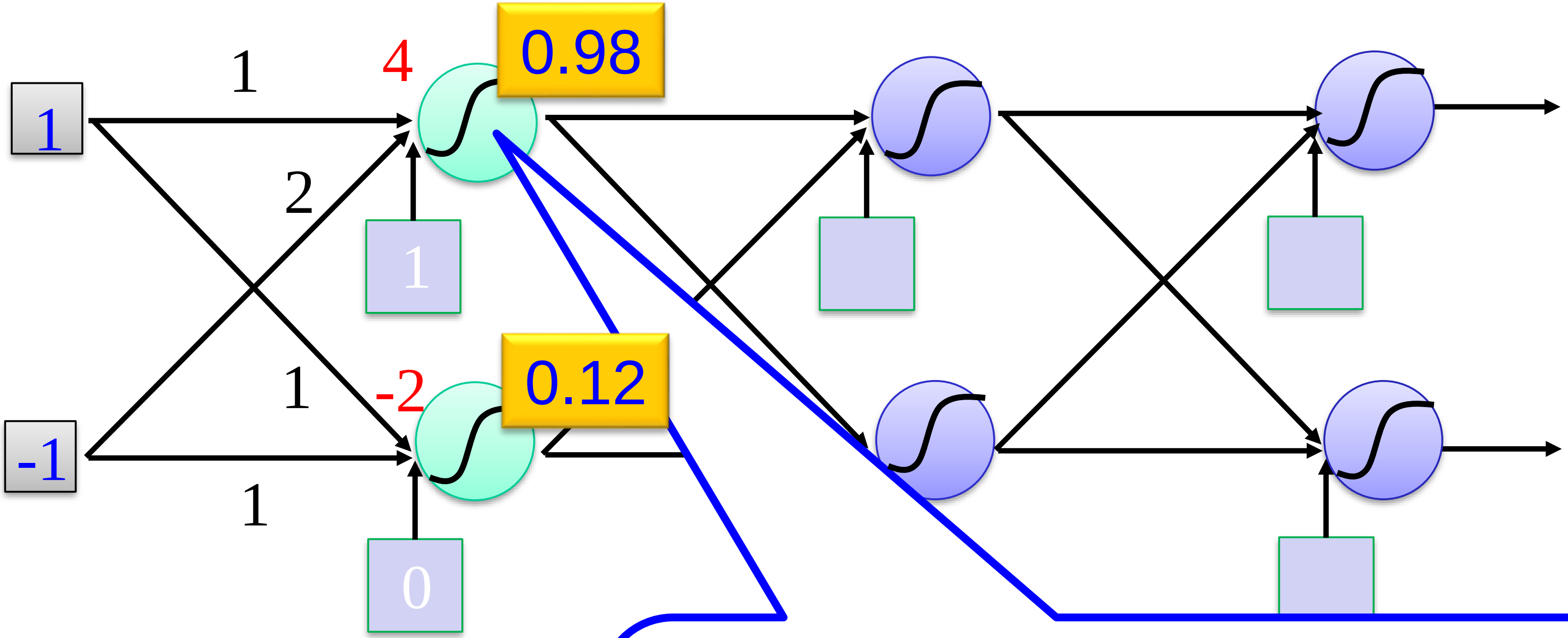
⋮

0.2 is 0

The image is "2"

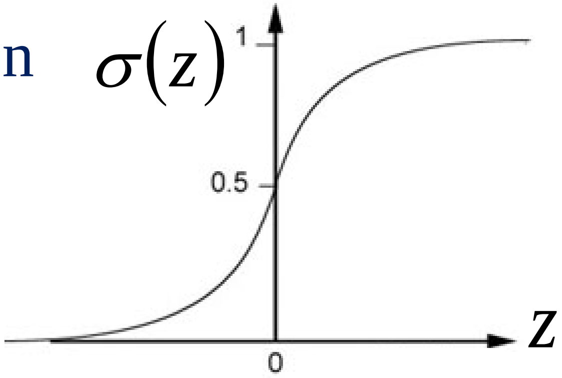
Each dimension represents the confidence of a digit.

Example of Neural Network

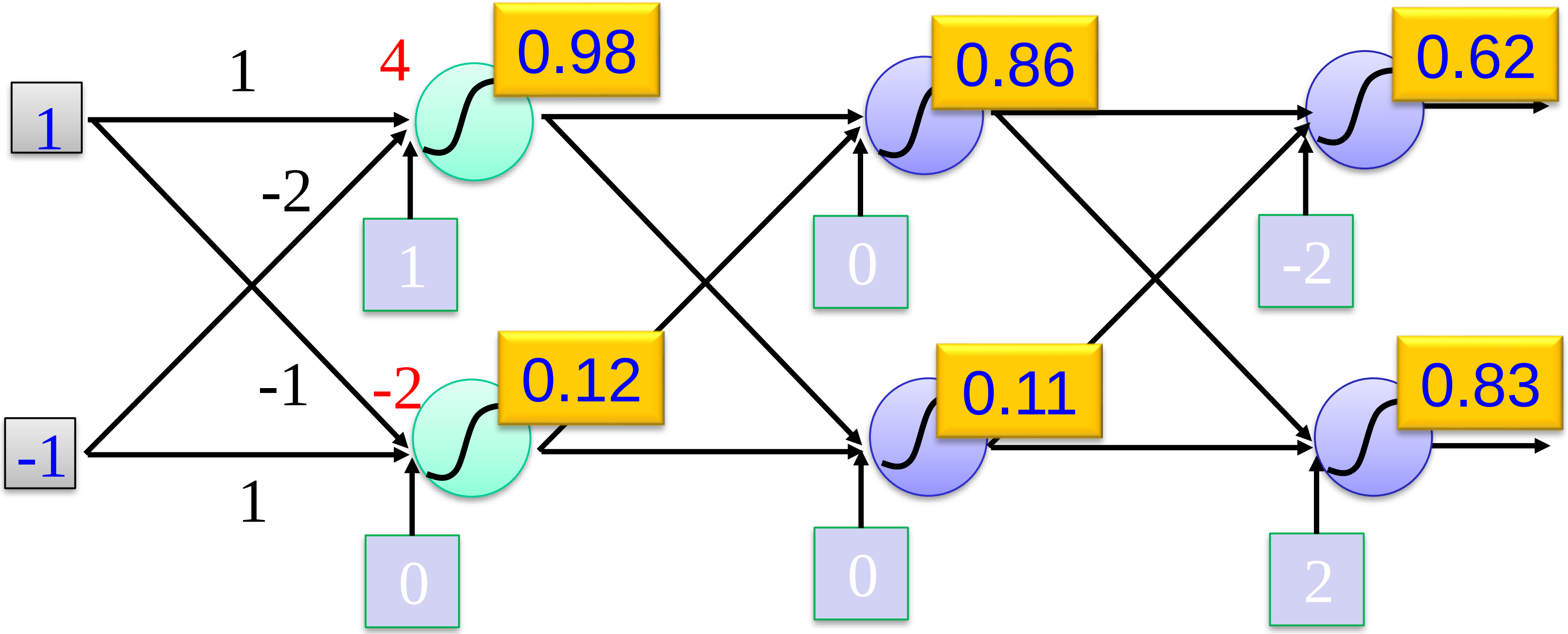


Sigmoid Function

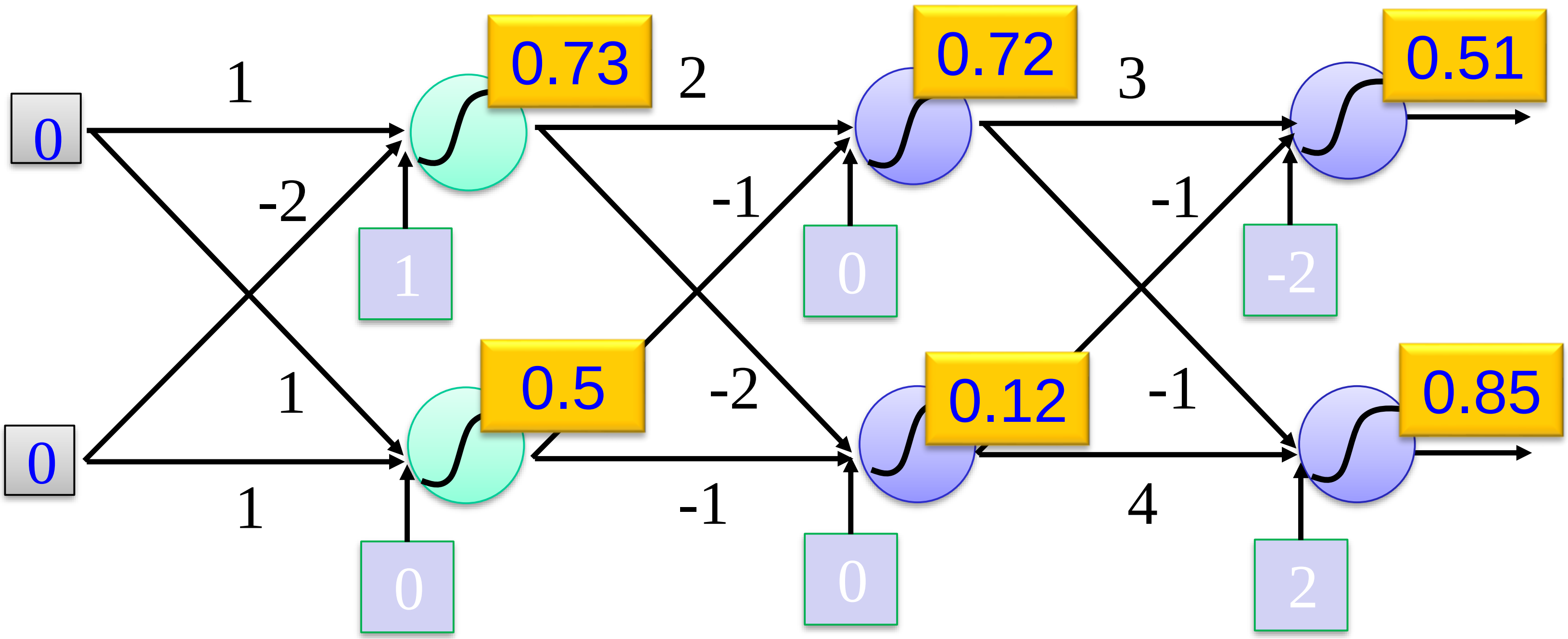
$$\sigma(z) = \frac{1}{1+e^{-z}}$$



Example of Neural Network



Example of Neural Network

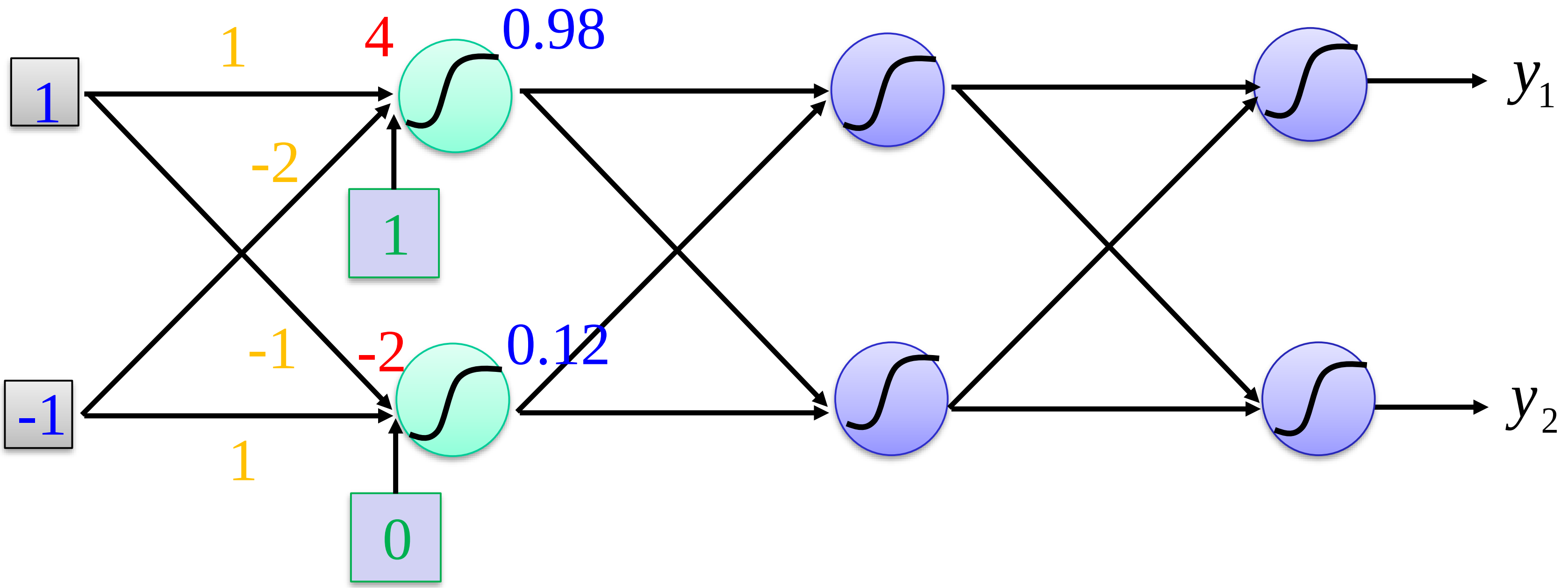


$f: R^2 \rightarrow R^2$

$$f\left(\begin{bmatrix}1\\-1\end{bmatrix}\right)=\begin{bmatrix}0.62\\0.83\end{bmatrix} \quad f\left(\begin{bmatrix}0\\0\end{bmatrix}\right)=\begin{bmatrix}0.51\\0.85\end{bmatrix}$$

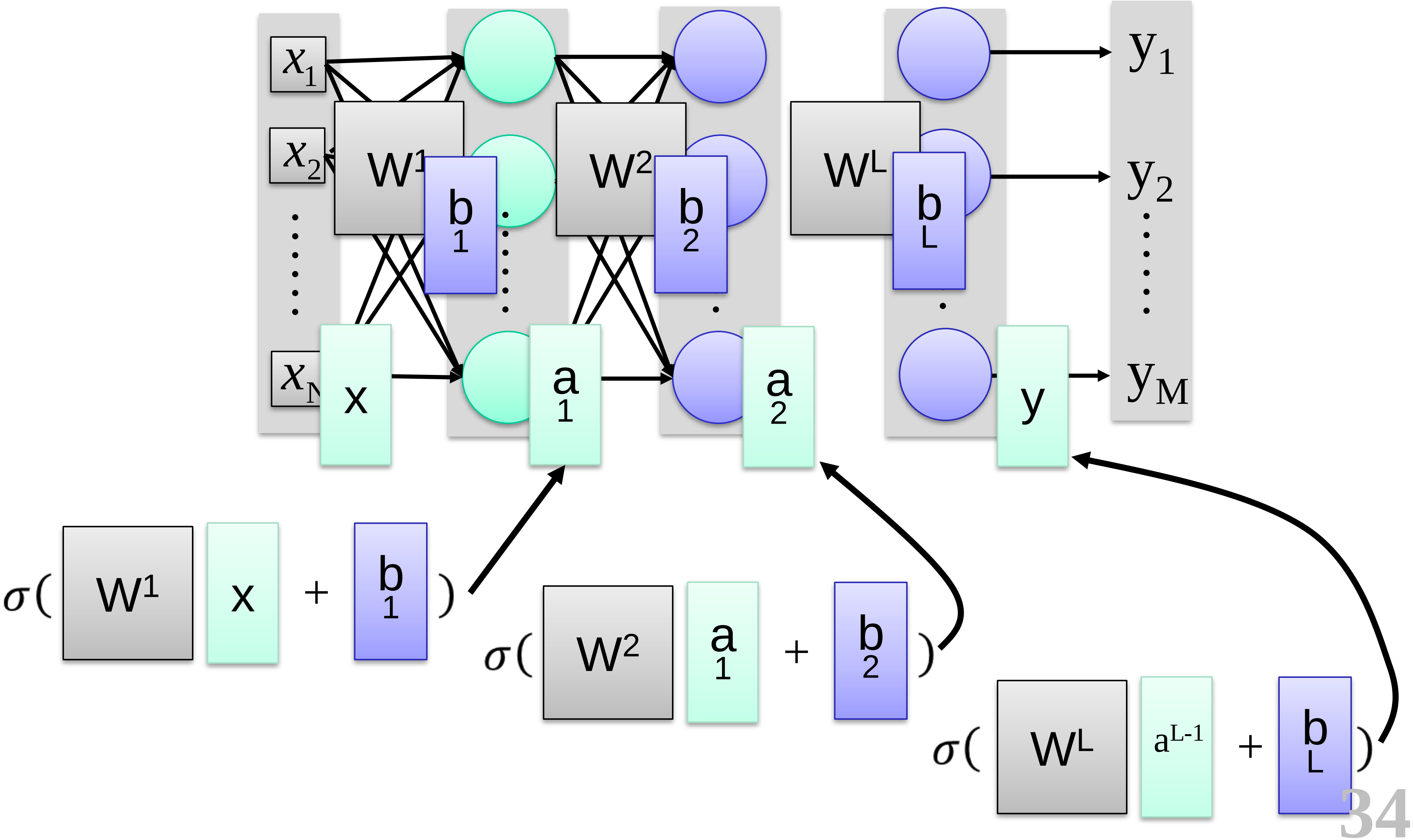
Different parameters define different function

Matrix Operation

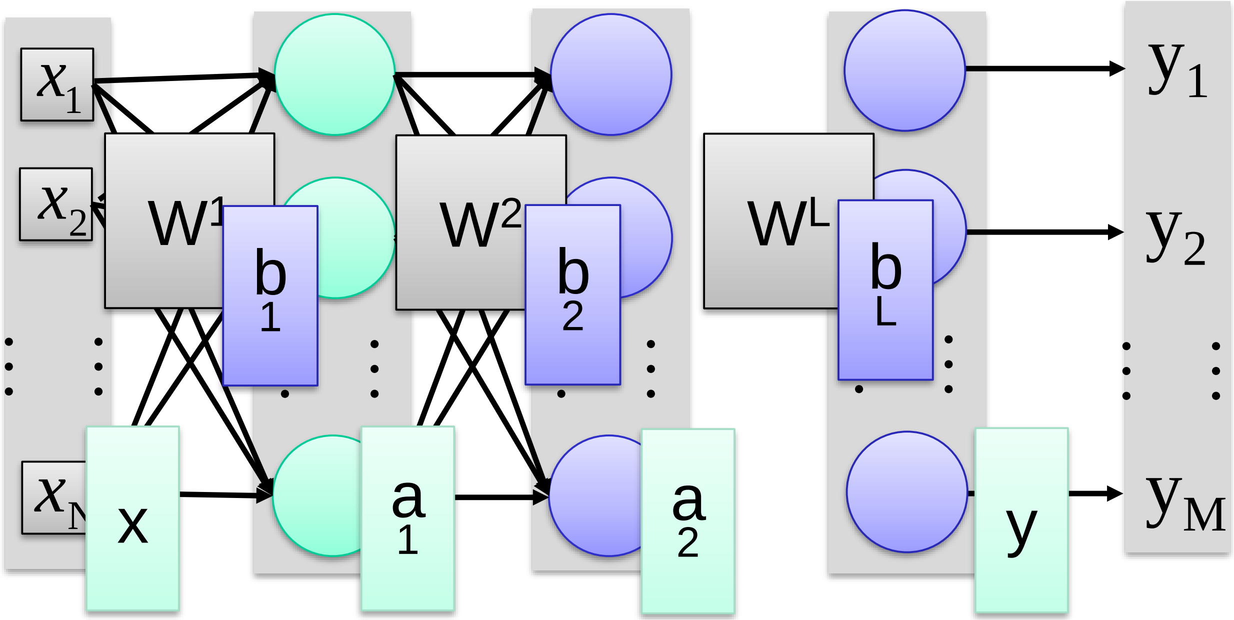


$$\sigma\left(\underbrace{\begin{bmatrix} 1 & -2 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad \begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{\begin{bmatrix} 4 \\ -2 \end{bmatrix}} \right) = \begin{bmatrix} 0.98 \\ 0.12 \end{bmatrix}$$

Neural Network



Neural Network



$y = f(x)$

Parallel computing techniques are needed to speed up **matrix operations**

$= \sigma(W^L \quad W^2 \quad W^1 \; x \quad b_1 \quad b_2 \quad b_L)$

DL is Computationally Expensive

- The two phases of NN are called *training* (or learning) and *inference* (or prediction), and they refer to development versus production.

- The Developer chooses the number of layers and the type of NN, and training determines the weights.

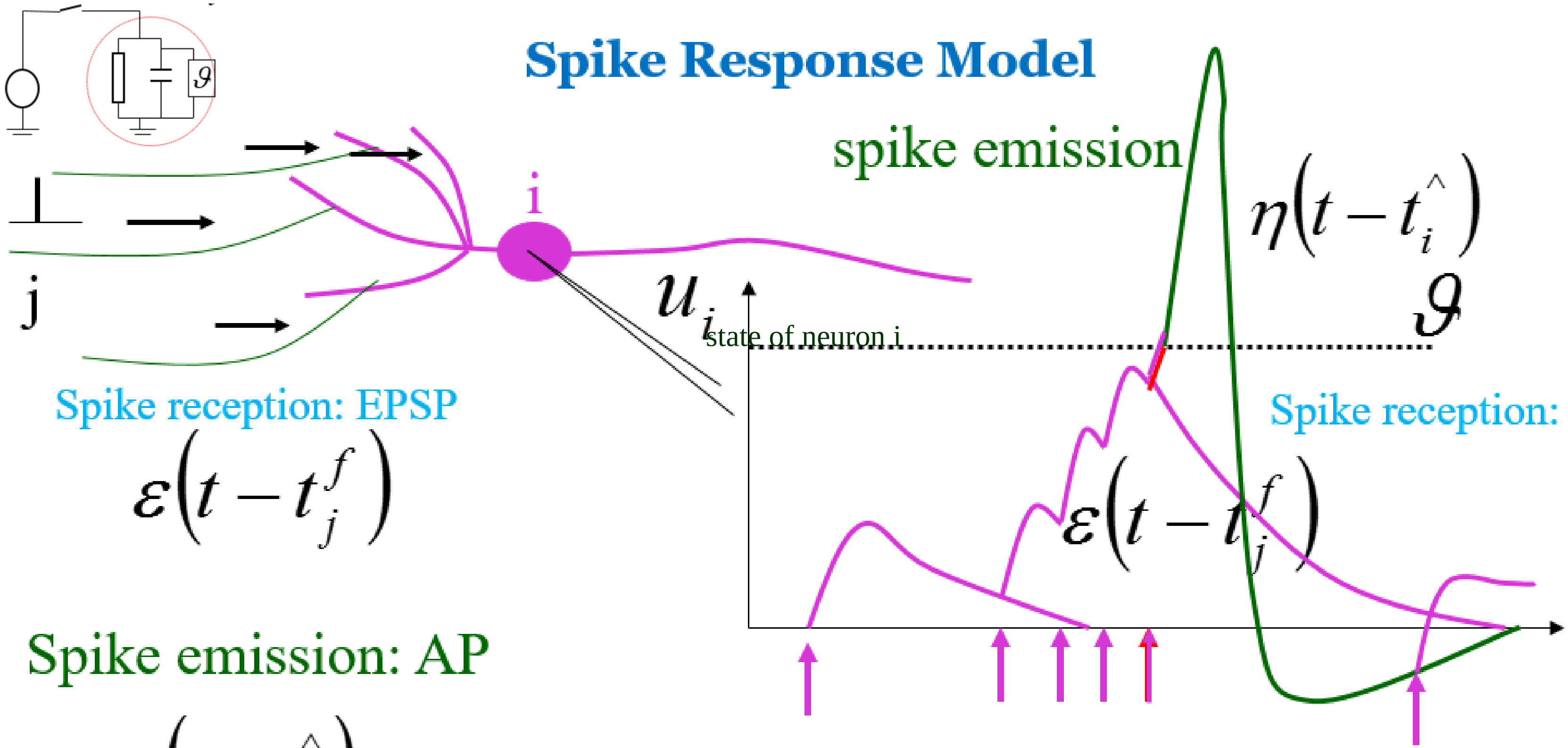
- Virtually all training today is in floating point.

- A step called ***quantization*** transforms floating-point numbers into narrow integers—often just 8 bits—which are usually good enough for inference.

- 8-bit integer multiplies can be 6X less energy and 6X less area than IEEE 754 16-bit FPMs, and the advantage for integer addition is 13X in energy and 38X in area [Dal16].

A more biological version: LIF/SRM Model

Spike Response Model



Spike reception: EPSP

$$\varepsilon(t - t_j^f)$$

Spike emission: AP

$$\eta(t - \hat{t}_i)$$

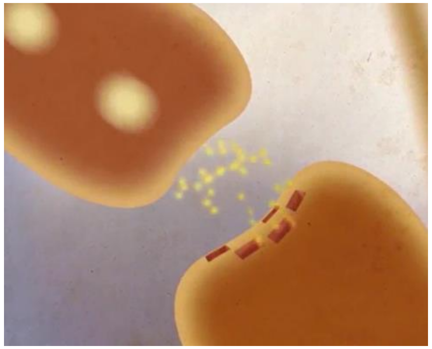
reset of the membrane potential (action potential)

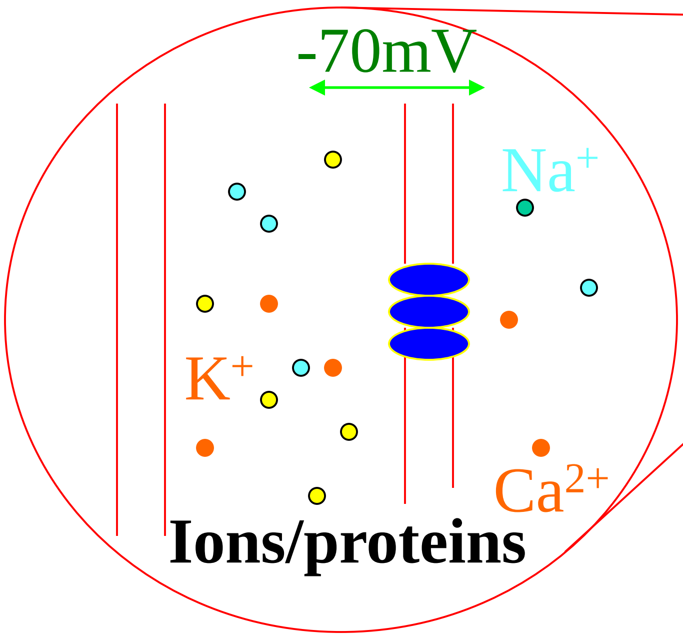
$$u_i(t) = \eta(t - \hat{t}_i) + \sum_j \sum_f w_{ij}\, \varepsilon(t - t_j^f)$$

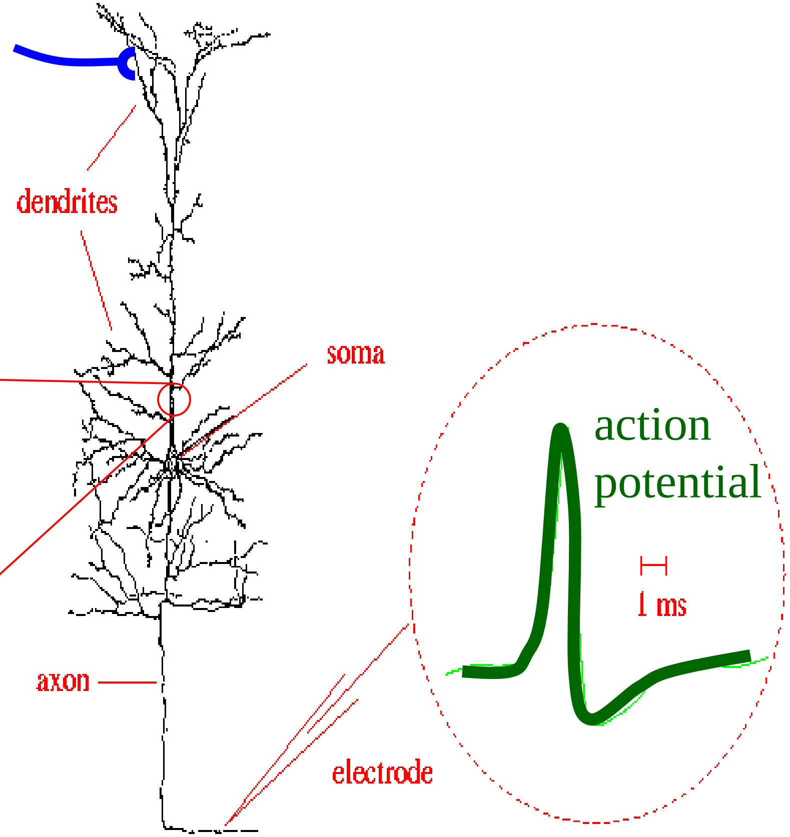
$$\boxed{u_i(t) = \vartheta \Rightarrow \text{Firing:} \quad \hat{t}_i = t}$$

[Ref. 18]

A more biological Model: Molecular Basis

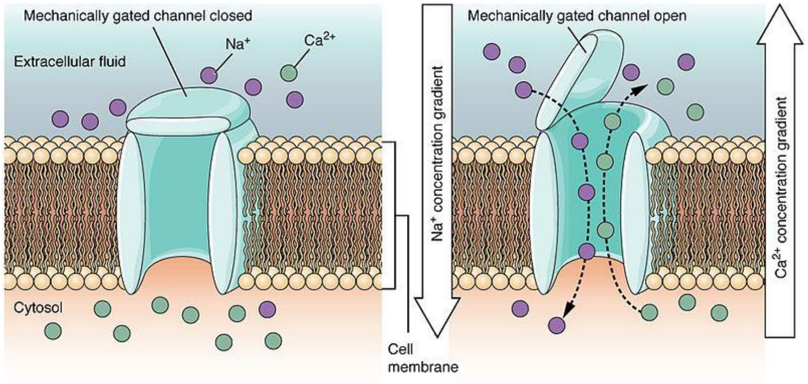


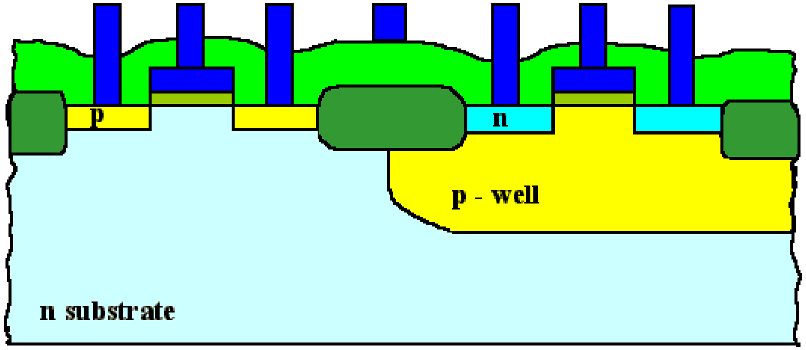




[Ref. 18]

Electronic devise vs chemical device

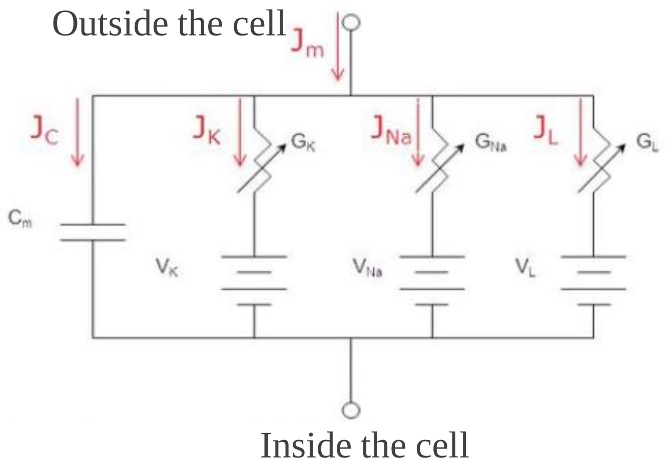

Mechanically gated channel closed
Extracellular fluid
Na+
Ca2+
Cytosol
Na+ concentration gradient
Cell membrane
Mechanically gated channel open
Ca2+ concentration gradient



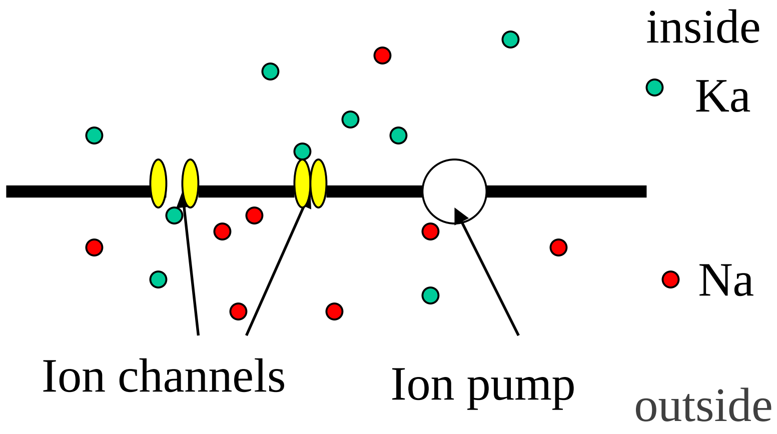
- Deliver the concentration difference of K+,Na+
- Action potential ~ 70 mV
  - ➢ Extreme low voltage operation
  - ➢ Noise problem
  - ➢ Multiple signal input/ integration

- Spatial and temporal multiplexing ➔ Active sharing of the interconnect
- Chemical computing, extremely low operation voltage (<100mV) ➔ Low power

Hodgkin-Huxley Model



~

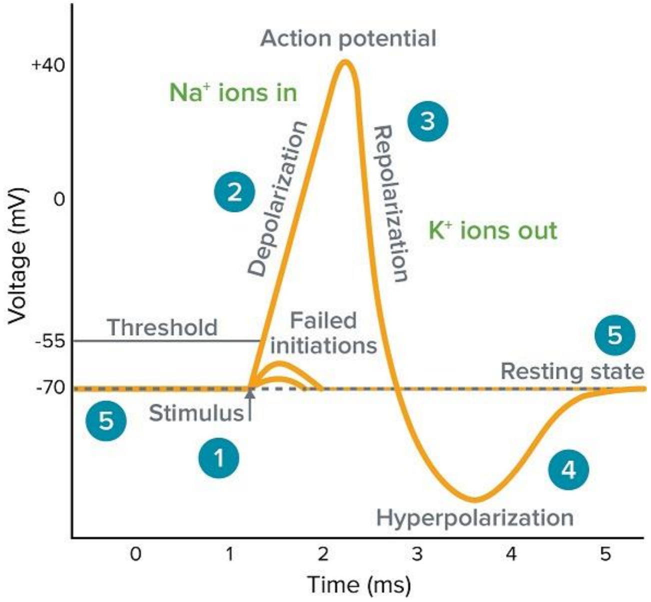


$$J_c = C_m \, {\partial V_m}/{\partial t}$$

$$J_{Na^+} = G_{Na^+}\left(V_m - V_{Na^+}\right)$$

$$J_{K^+} = G_{K^+}\left(V_m - V_{K^+}\right)$$

$$J_L = G_L\left(V_m - V_L\right)$$

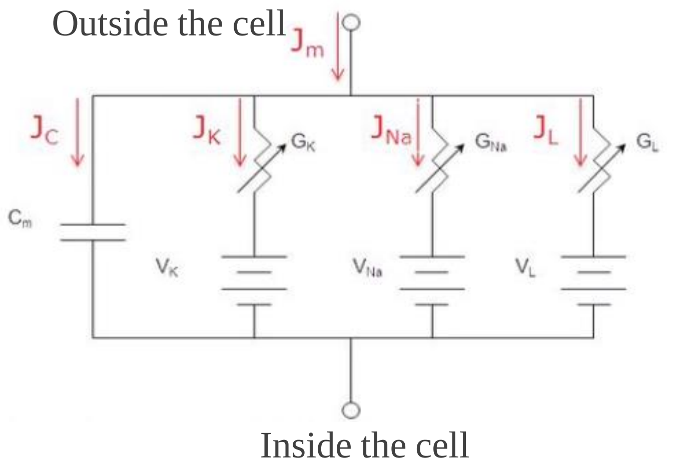


$$J_m = J_c + J_{K^+} + J_{Na^+} + J_L$$

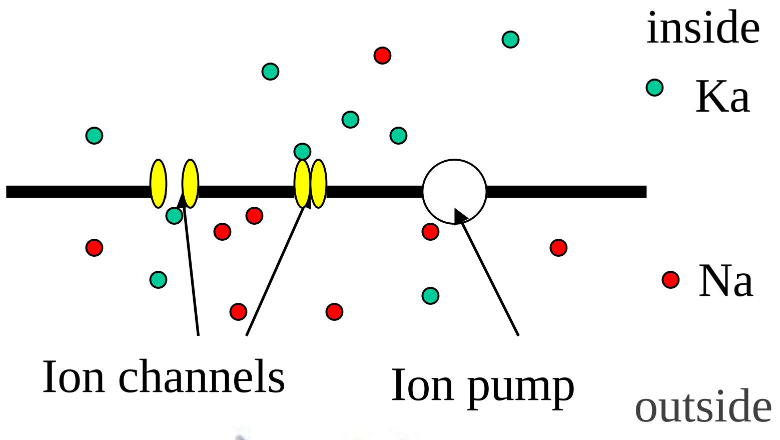
$$J_m = C_m \, {\partial V_m}/{\partial t} + G_{K^+}\left(V_m - V_{K^+}\right) + G_{Na^+}\left(V_m - V_{Na^+}\right) + G_L\left(V_m - V_L\right)$$

[Ref. 18]

Hodgkin-Huxley Model



~

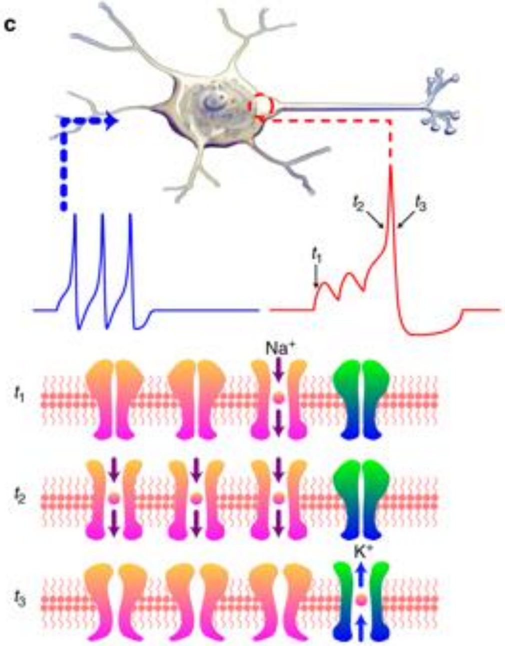


$$J_c = C_m \, {}^{\partial V_m}\!/_{\partial t}$$

$$J_{Na^+} = G_{Na^+}\left(V_m - V_{Na^+}\right)$$

$$J_{K^+} = G_{K^+}\left(V_m - V_{K^+}\right)$$

$$J_L = G_L\left(V_m - V_L\right)$$



$$J_m = J_c + J_{K^+} + J_{Na^+} + J_L$$

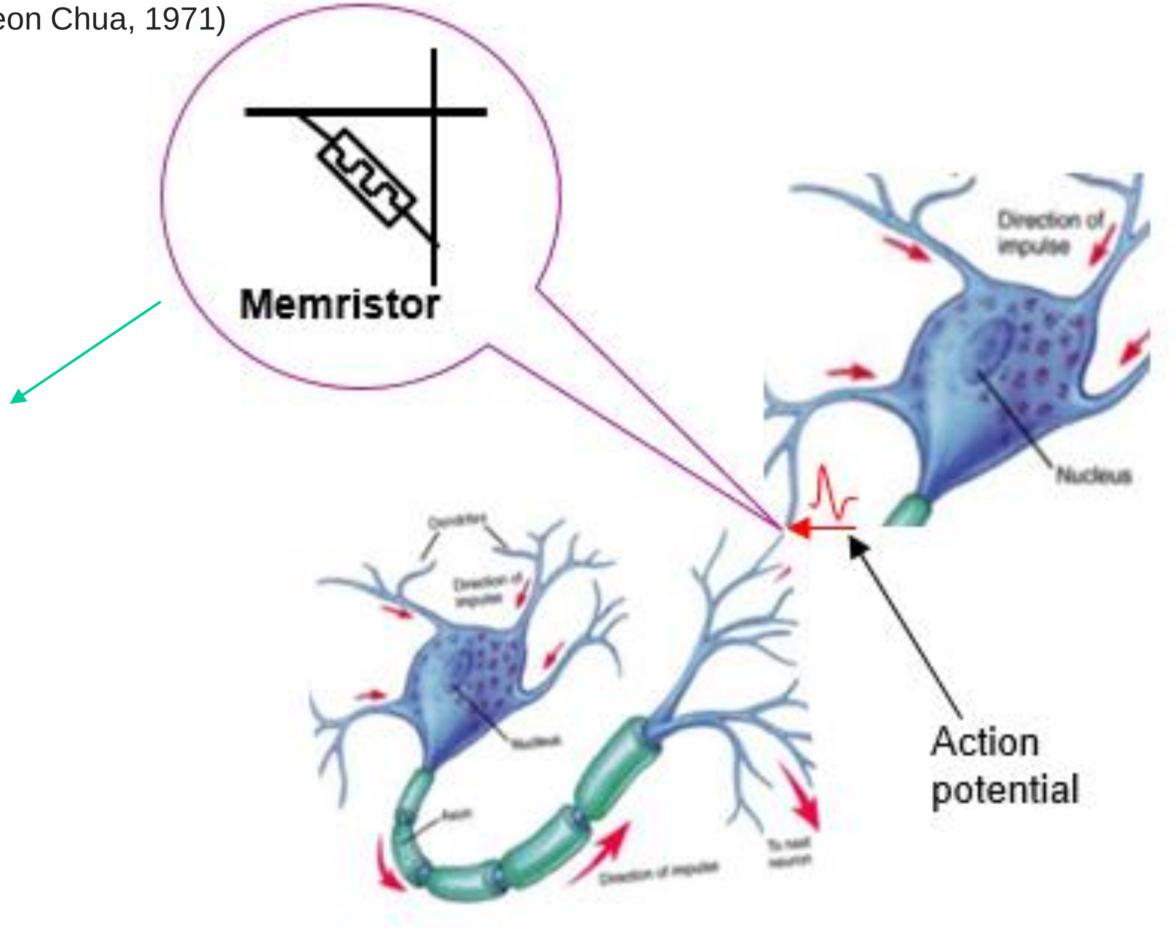
$$J_m = C_m \, {}^{\partial V_m}\!/_{\partial t} + G_{K^+}\left(V_m - V_{K^+}\right) + G_{Na^+}\left(V_m - V_{Na^+}\right) + G_L\left(V_m - V_L\right)$$

[Ref. 18]

Action Potential (Synapse) Storage

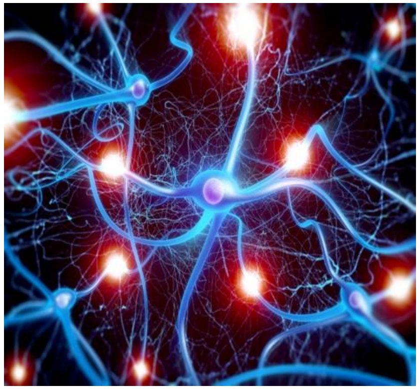
(Dr. Leon Chua, 1971)

The electrical resistor is not constant but depends on the history of current that had previously flowed through the device.

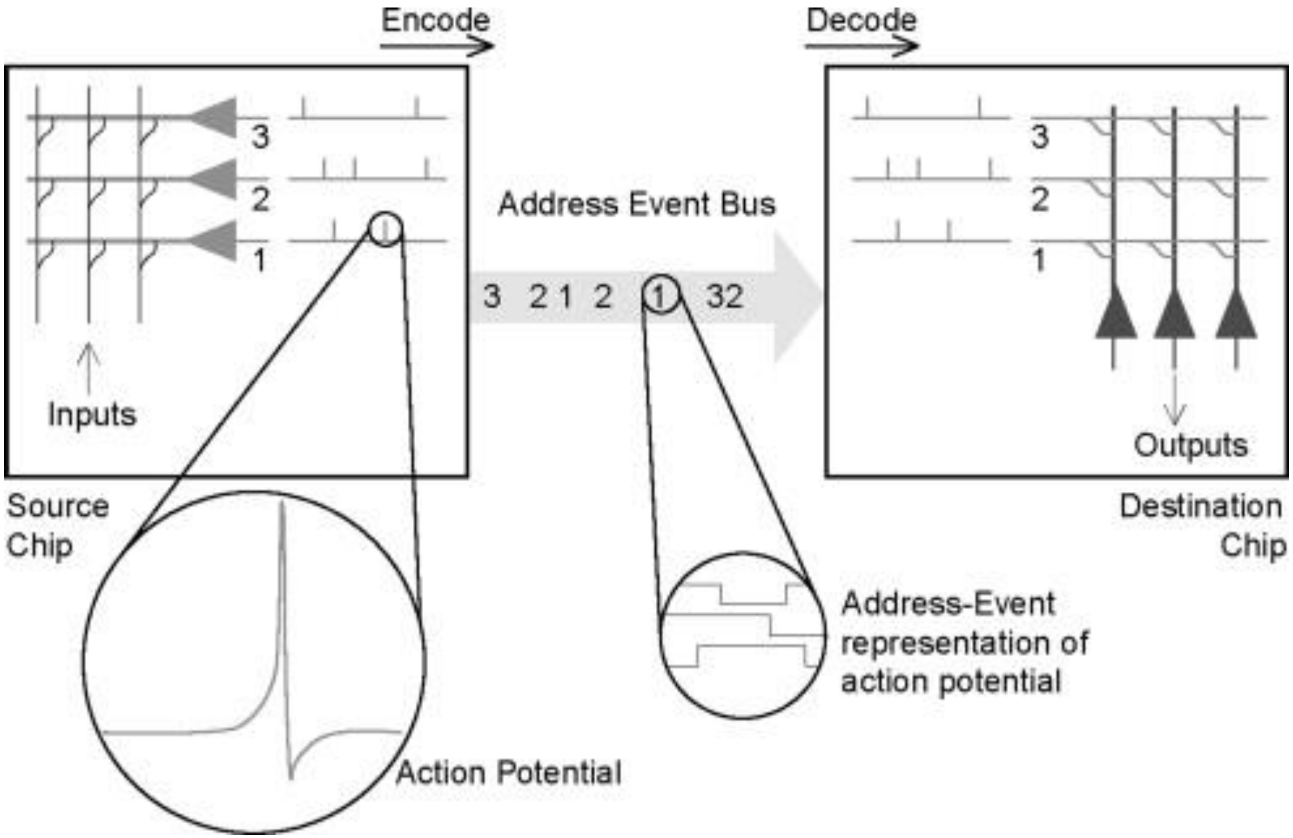


- ❖Voltage **pulses** can be applied to a **memristor** to change its **resistance**, just as **spikes** can be applied to a **synapse** to change its **weight**.

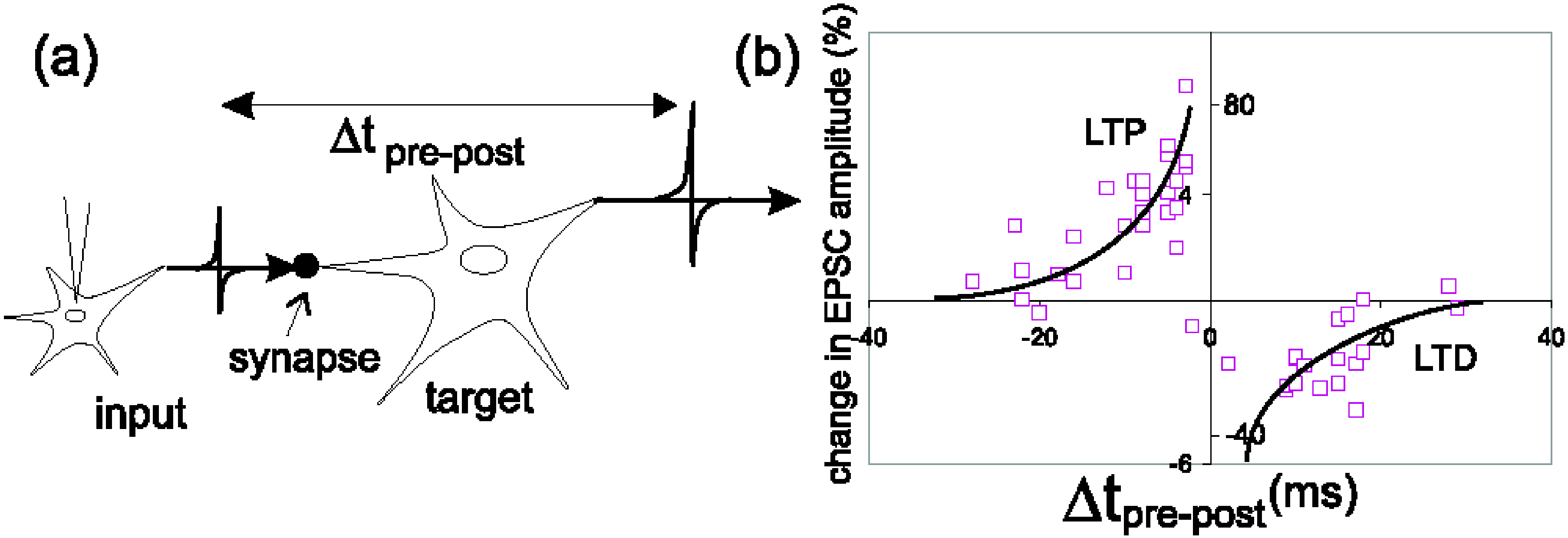
Wiring via AER – address Event Representation



(Courtesy: iStock/Henrik5000)



Spike-timing-dependent plasticity (STDP)

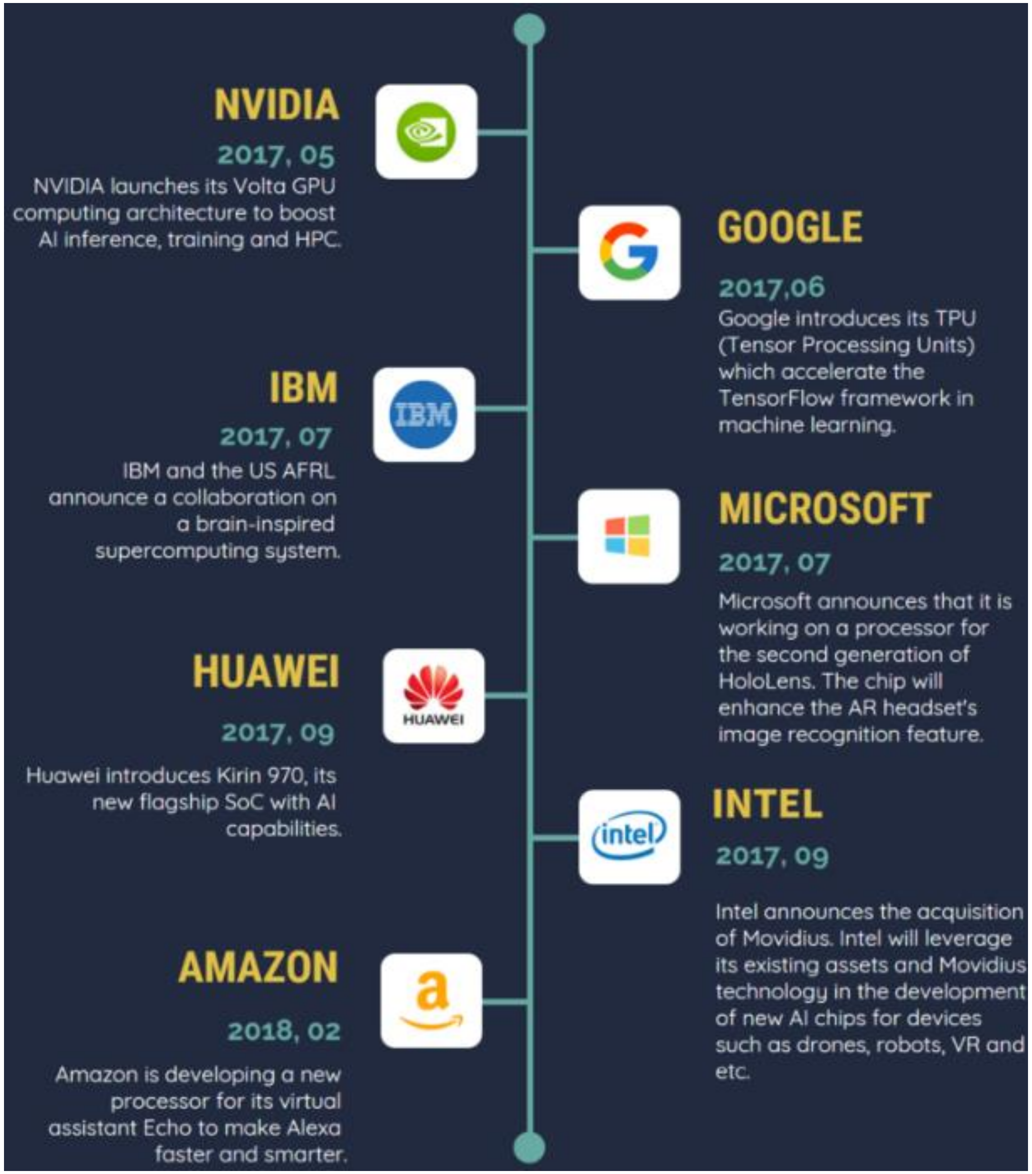


- Adjusts the strength of connections between neurons in the brain.
  - ✓ Adjusts the connection strengths based on the relative timing of a particular neuron's output and input action potentials.

Agenda

- Fundamental Trends
- AI – The 4th Industrial Revolution
- Survey of AI Hardware
  - ➢ Cloud AI Hardware, Chips
  - ➢ Mobile AI Chips
  - ➢ Edge and IoT AI Chips
  - ➢ Healthcare AI Chips
- Conclusions

Big Corps AI Chips



NVIDIA

2017, 05

NVIDIA launches its Volta GPU computing architecture to boost AI inference, training and HPC.

GOOGLE

2017,06

Google introduces its TPU (Tensor Processing Units) which accelerate the TensorFlow framework in machine learning.

IBM

2017, 07

IBM and the US AFRL announce a collaboration on a brain-inspired supercomputing system.

MICROSOFT

2017, 07

Microsoft announces that it is working on a processor for the second generation of HoloLens. The chip will enhance the AR headset's image recognition feature.

HUAWEI

2017, 09

Huawei introduces Kirin 970, its new flagship SoC with AI capabilities.

INTEL

2017, 09

Intel announces the acquisition of Movidius. Intel will leverage its existing assets and Movidius technology in the development of new AI chips for devices such as drones, robots, VR and etc.

AMAZON

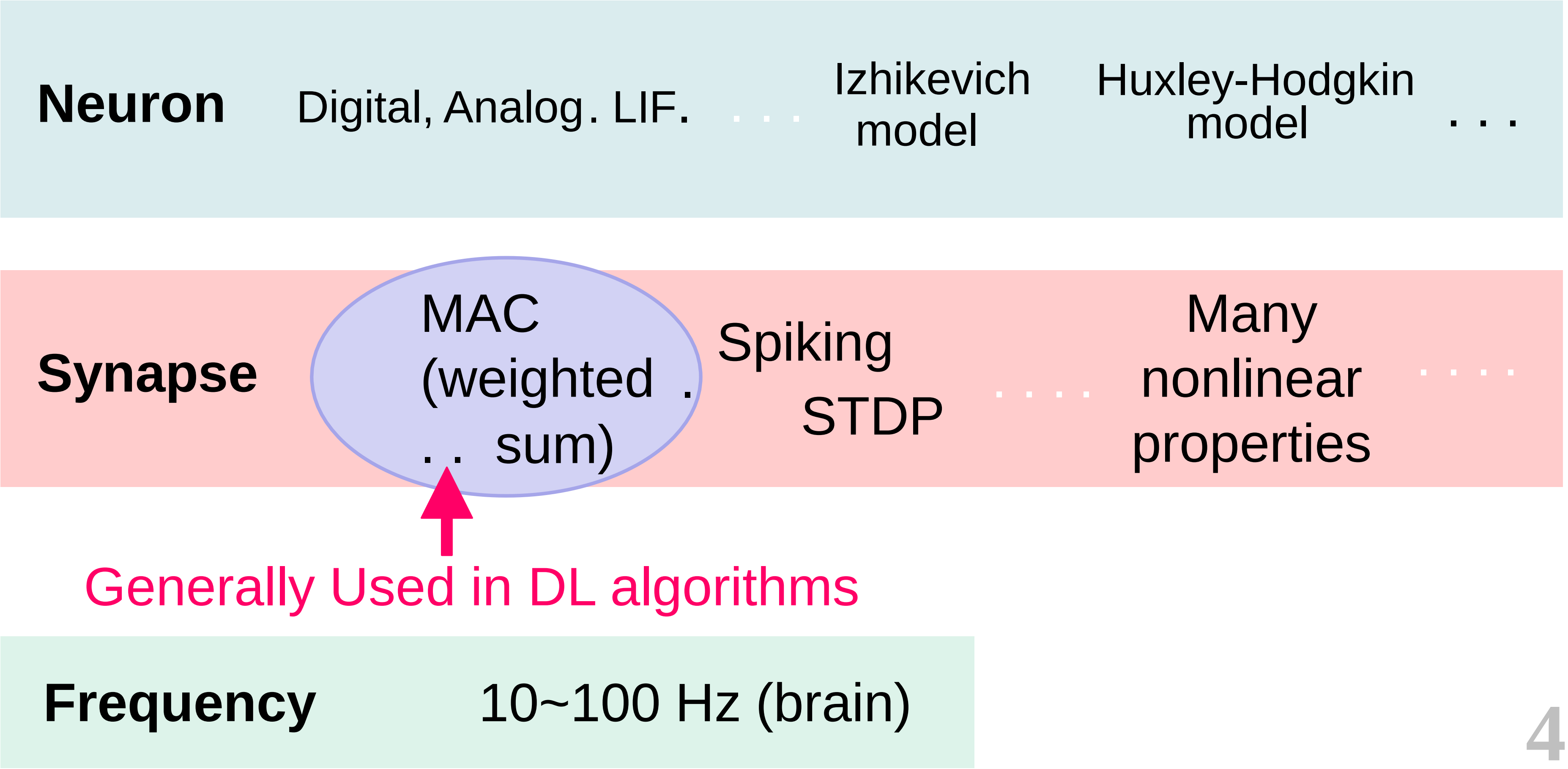
2018, 02

Amazon is developing a new processor for its virtual assistant Echo to make Alexa faster and smarter.

[source: medium.com]

The are two AI Chip Models: ANN and SNN

- The output of ANN Chip depends only on the current stimuli, the output of SNN depends on previous stimuli also
- The SNN/Neuromorphic Chip operates on biology-inspired principles to improve performance and increase energy efficiency



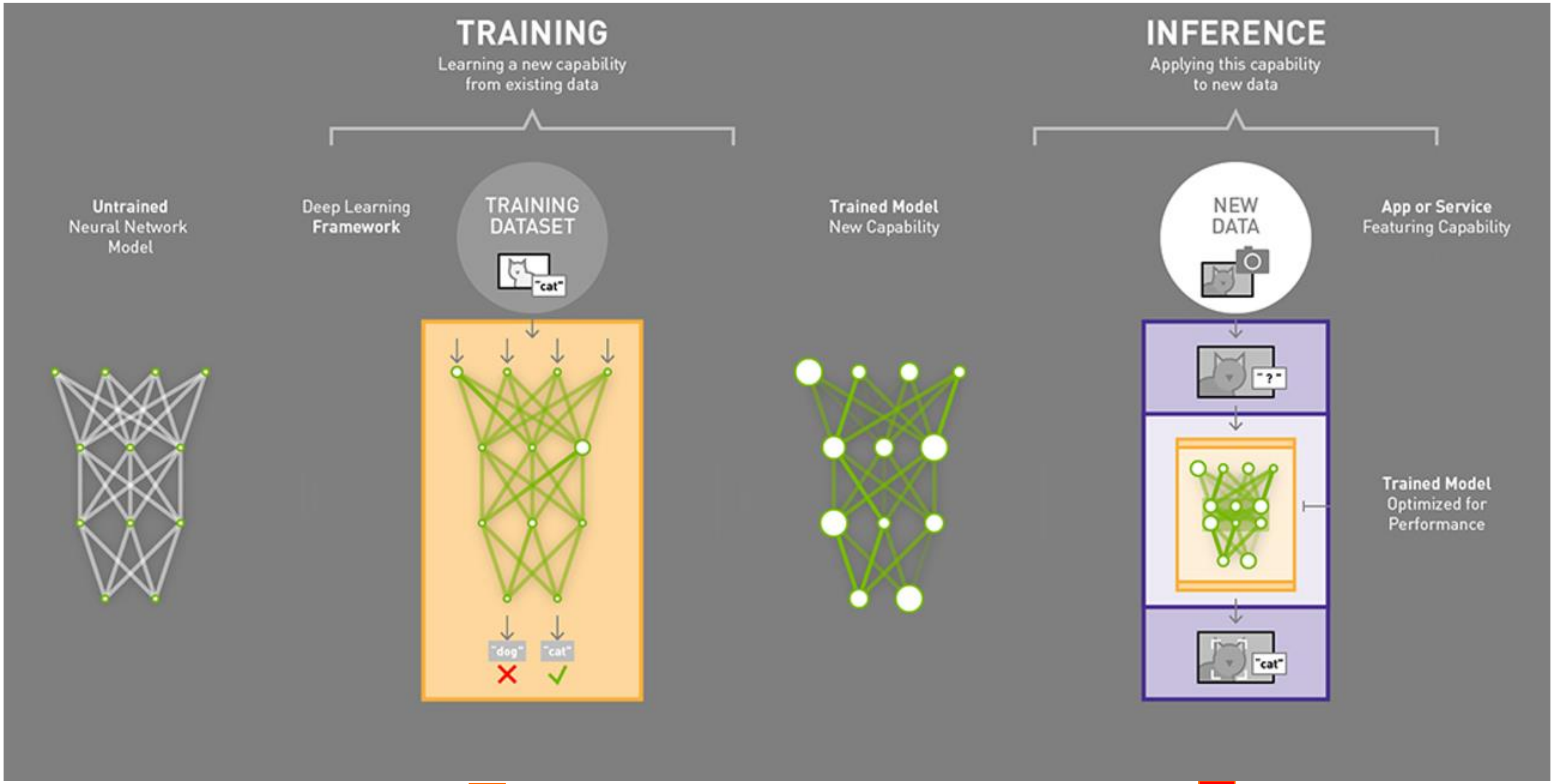
Neuron Digital, Analog. LIF. . . . Izhikevich model Huxley-Hodgkin model . . .

Synapse MAC (weighted . . . sum) Spiking STDP . . . . Many nonlinear properties . . . . .

Generally Used in DL algorithms

Frequency 10~100 Hz (brain)

Training & Inference


TRAINING
Learning a new capability from existing data
INFERENCE
Applying this capability to new data
Untrained Neural Network Model
Deep Learning Framework
TRAINING DATASET
"cat"
"dog"
"cat"
Trained Model New Capability
NEW DATA
App or Service Featuring Capability
"?"
Trained Model Optimized for Performance
"cat"

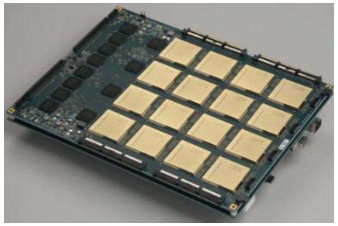
FPGA, GPU, Cloud

CPU, FPGA, GPU, ASIC

Neuromorphic/SNN AI-Chips

- **Neuromorphic Sensors** - electronic models of retinas and cochleas.
- **Smart sensors –** tracking chips, motion, pressor, auditory classifications and localization sensors.
- **Models of specific systems**: e.g. lamprey spinal cord for swimming, electric fish lateral line.
- **Pattern generators** – for locomotion or rhythmic behavior
- **Large-scale multi-core/chip systems** – for investigating models of neuronal computation and synaptic plasticity.

Neurogrid (Stanford)

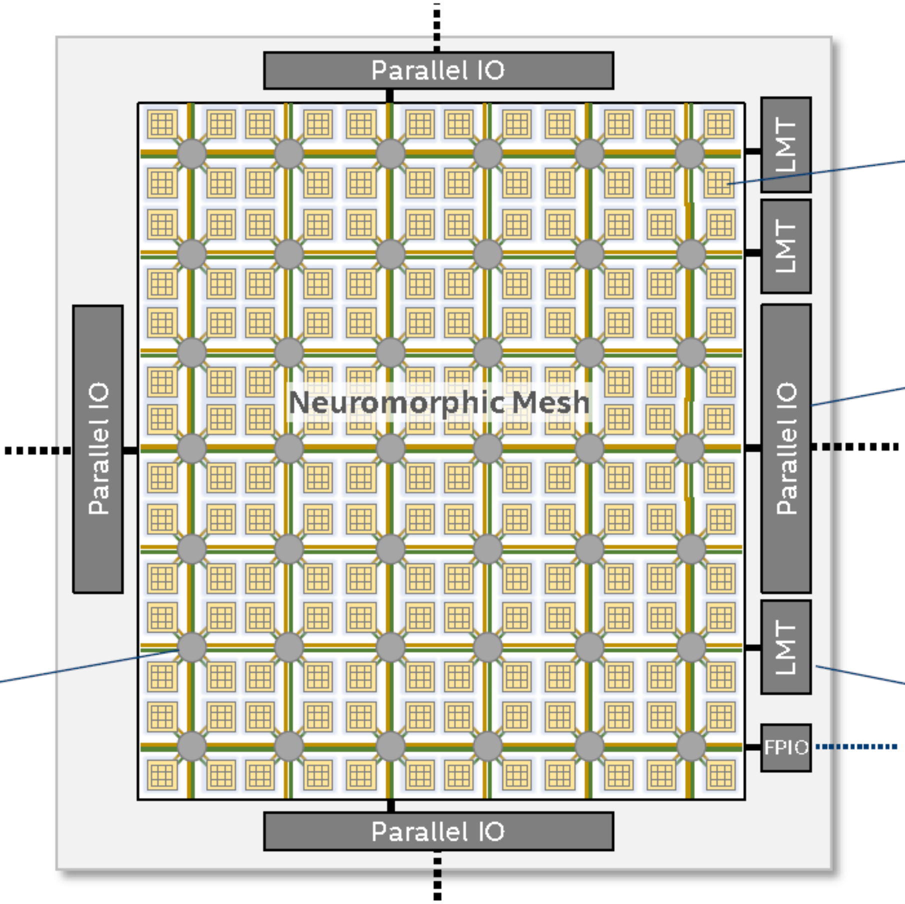

TrueNorth (IBM)

Brainscales/HBP (Heidelberg, Lausanne)

SpiNNaker (Manchester)

Example
Loihi AI-Chip - a 60-mm2 chip fabricated in Intel’s 14-nm

Technology:	14nm
Die Area:	64 mm²
Core area:	0.41 mm²
NmC cores:	128 cores
x86 cores:	3 LMT cores
Max # neurons:	128K neurons
Max # synapses:	128M synapses
Transistors:	2.07 billion



Neuromorphic core
- LIF neuron model
- Programmable learning
- 128 KB synaptic memory
- Up to 1,024 neurons
- Asynchronous design

Parallel off-chip interfaces
- Two-phase asynchronous
- Single-ended signaling
- 100-200 MB/s BW

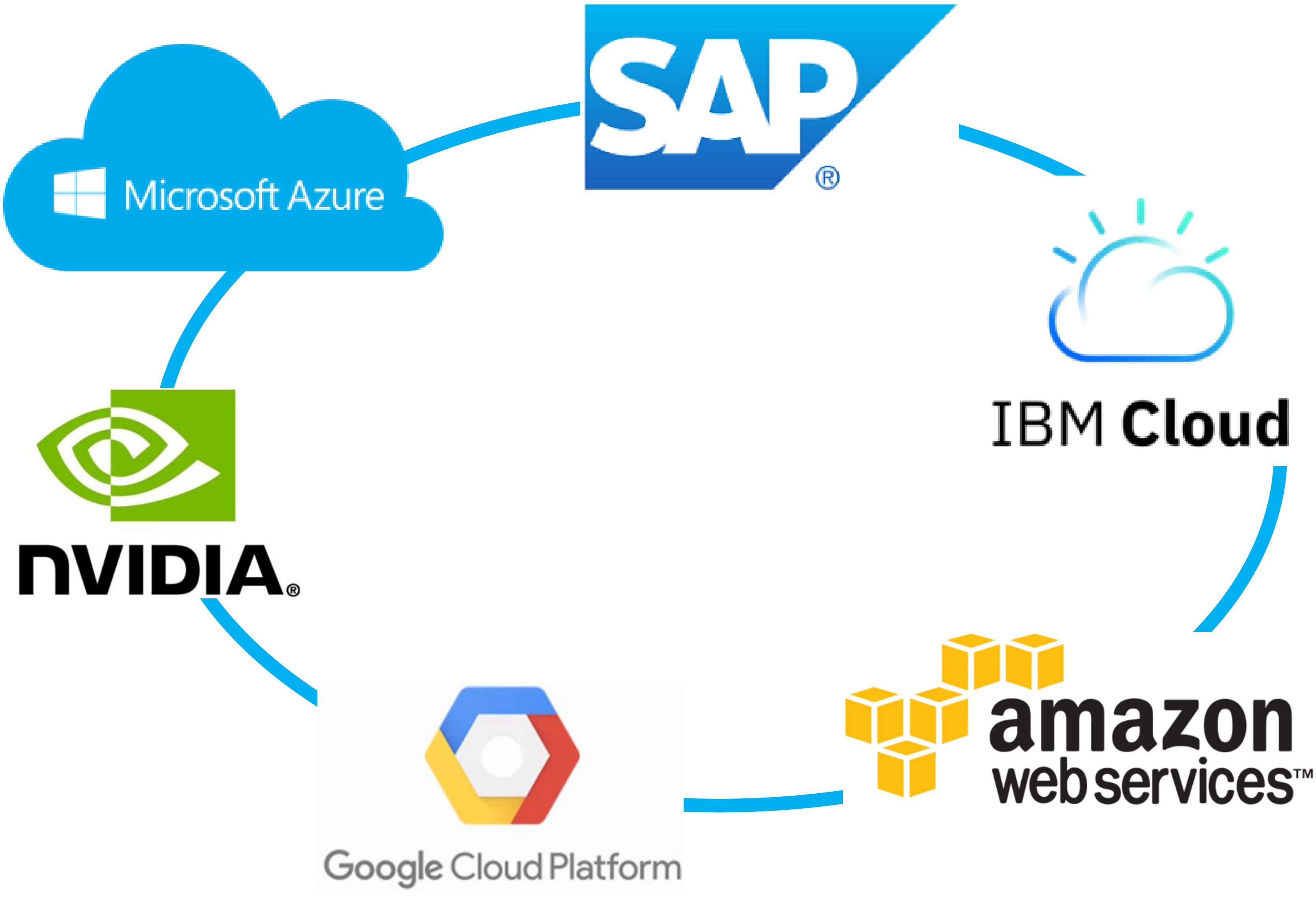
Embedded x86 processors
- Efficient spike-based communication with neuromorphic cores
- Data encoding/decoding
- Network configuration
- Synchronous design

Low-overhead NoC fabric
- 8x16-core 2D mesh
- Scalable to 1000’s cores
- Dimension order routed
- Two physical fabrics
- 8 GB/s per hop

M. Davieset *al*., “Loihi: A Neuromorphic Manycore Processor with On-Chip Learning,” *IEEE Micro*, vol. 38, no. 1, pp. 82-99, January/February 2018

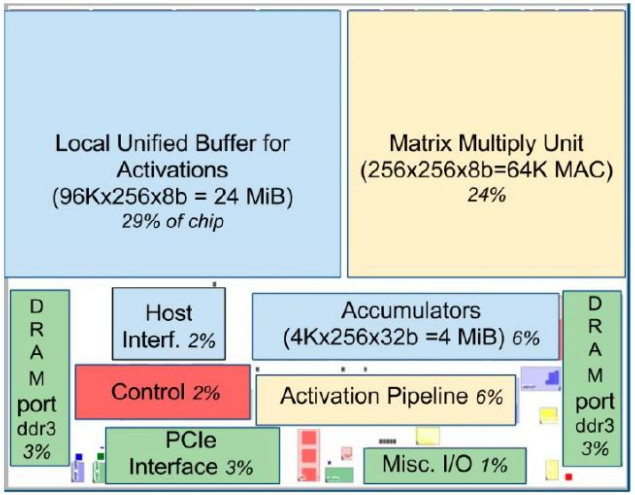
Cloud AI-Chips

Acceleration enterprise AI with DL Cloud



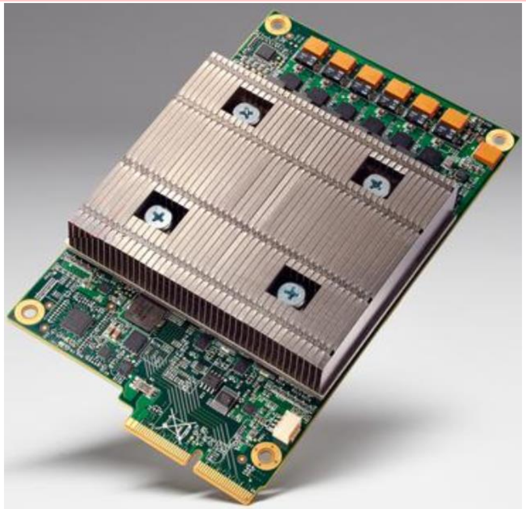
Custom ASIC: Tensor Processing Unit (TPU)

TPU is deployed in datacenters since 2015 that accelerates the inference phase of neural networks (NNs).

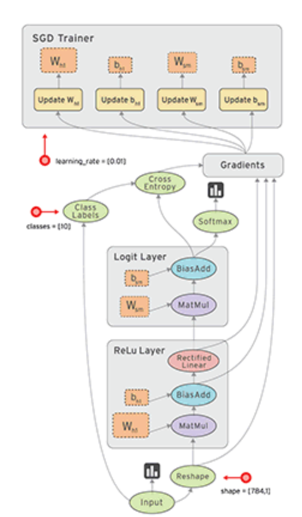


Local Unified Buffer for Activations (96Kx256x8b = 24 MiB) *29% of chip*
Matrix Multiply Unit (256x256x8b=64K MAC) *24%*
D R A M port ddr3 3%
Host Interf. 2%
Accumulators (4Kx256x32b =4 MiB) 6%
D R A M port ddr3 3%
Control 2%
Activation Pipeline 6%
PCIe Interface 3%
Misc. I/O 1%

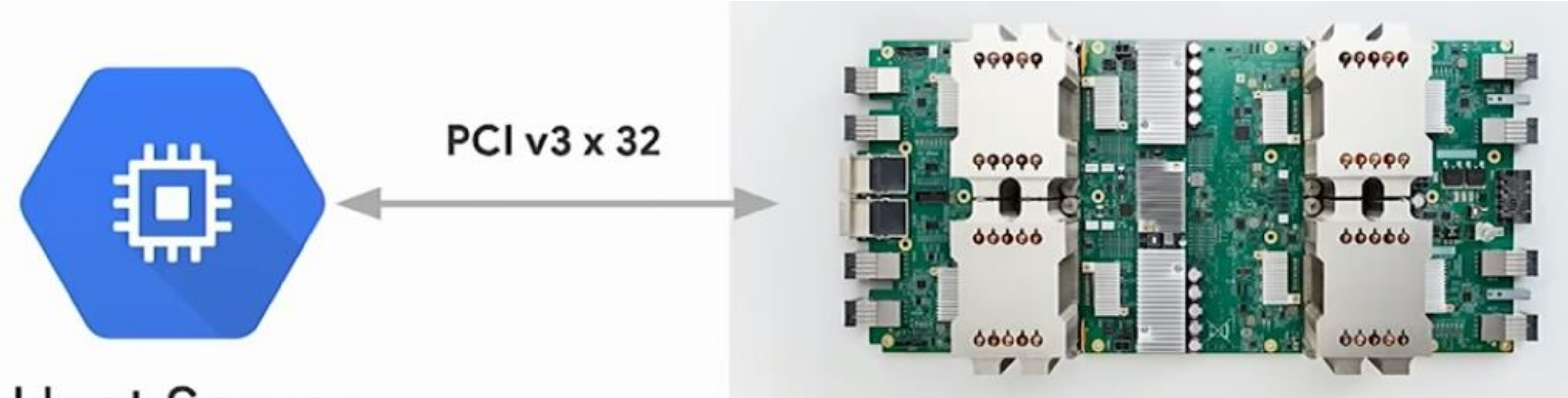
Floor Plan of TPU die



TPU Printed Circuit Board



Source: TensorFlow.org

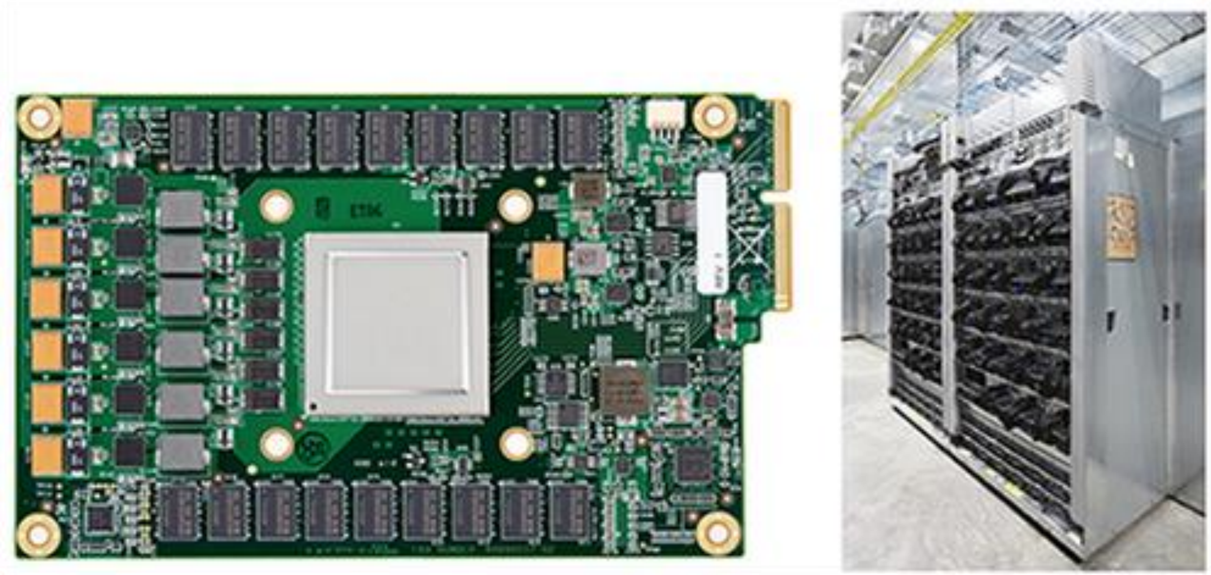


TPU chip runs at only 700 MHz

Source: In-datacenter Performance Analysis of a Tensor Processing Unit Jouppi et al, ISCA, 6/2017

Custom ASIC: Tensor Processing Unit (TPU)

TPU is deployed in datacenters since 2015 that accelerates the inference phase of neural networks (NNs).



Google’s first Tensor Processing Unit (TPU) on a printed circuit board (left); TPUs deployed in a Google datacenter (right)

Source: cloud.google.com

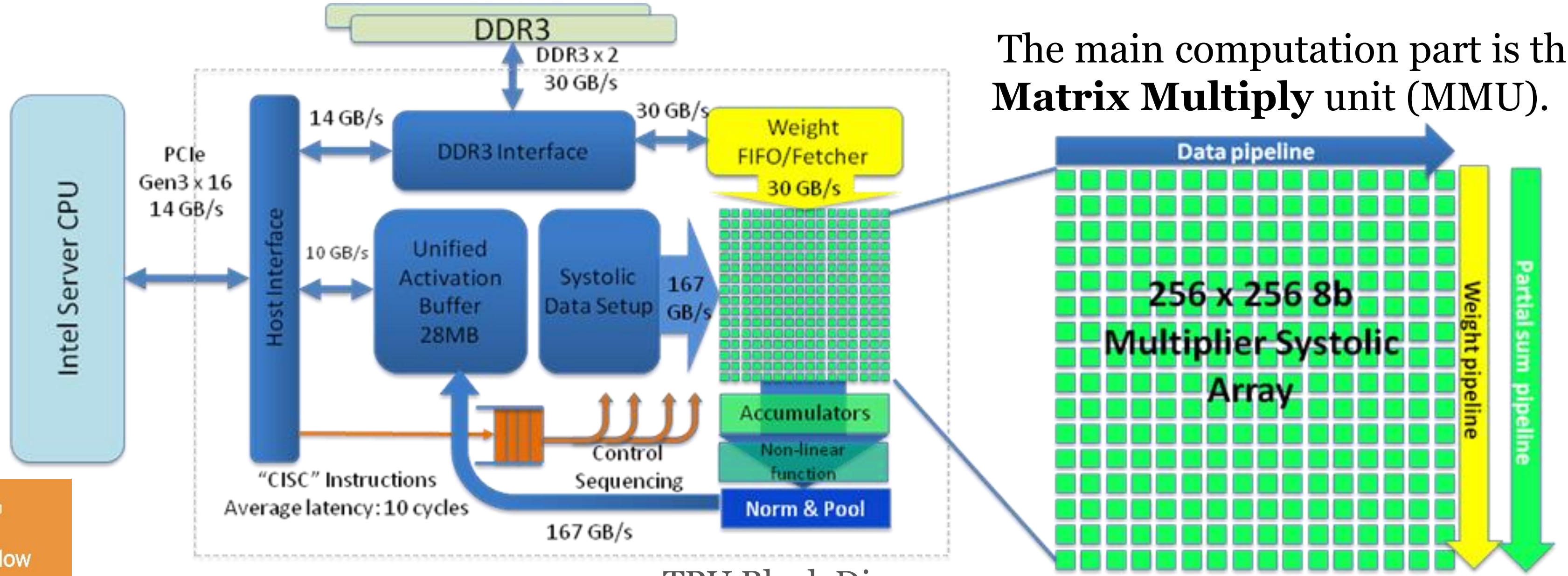
- The TPU board can perform 92 TeraOps/s (TOPS). It is 15 to 30 times faster than CPUs and GPUs tasked with the same work, with a 30- to 80-fold improvement in TOPS/W.
- The software used for comparison of systems was the TensorFlow framework.

Experience Cloud TPU: https://github.com/tensorflow/tpu
https://cloud.google.com/tpu/docs

Source: In-datacenter Performance Analysis of a Tensor Processing Unit Jouppi et al, ISCA, 6/2017

Custom ASIC: Tensor Processing Unit (TPU)

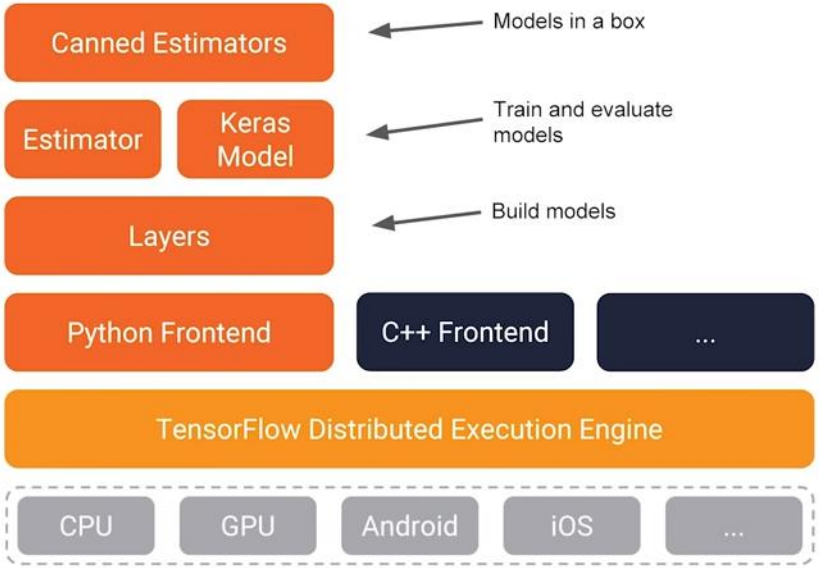
The main computation part is the **Matrix Multiply** unit (MMU).



TPU Block Diagram

Figure source: semiengineering.com

TensorFlow



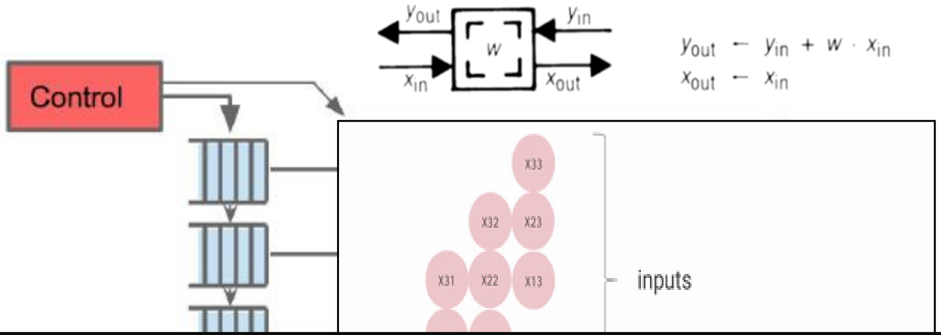
TensorFlow Platform Layers

- The portion of the application run on the TPU is typically written in **TensorFlow** and is compiled into an API that can run on GPUs or TPUs.

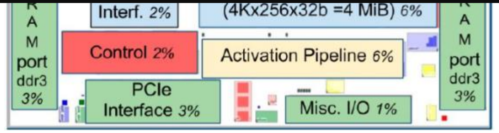
✓ **The TPU has a CISC-like instructions set:**
  - ✓ **Read_Host**
  - ✓ **Read_Weights**
  - ✓ **MatrixMultiply/Convolve**
  - ✓ **Activate**
  - ✓ **Write_Host**

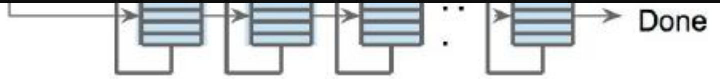
TPU is based on the Systolic Array Idea

The matrix unit uses systolic execution to save energy by reducing reads and writes of the **Unified Buffer.**


y_{out}
y_{in}
w
x_{in}
x_{out}
$y_{out} \leftarrow y_{in} + w \cdot x_{in}$
$x_{out} \leftarrow x_{in}$
Control
X33
X32
X23
X31
X22
X13
inputs

Benefit: Maximizes computation done on a single piece of data element brought from memory.





Systolic data flow of the Matrix Multiply Unit. SW has the illusion that each 256B input is read at once, and they instantly update one location of each of 256 accumulator RAMs.

TPU is based on the Systolic Array

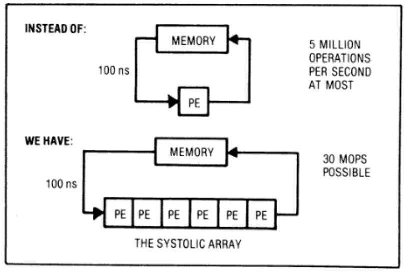
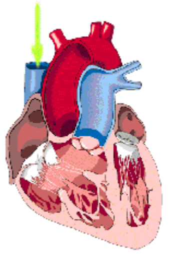


Figure 1. Basic principle of a systolic system.



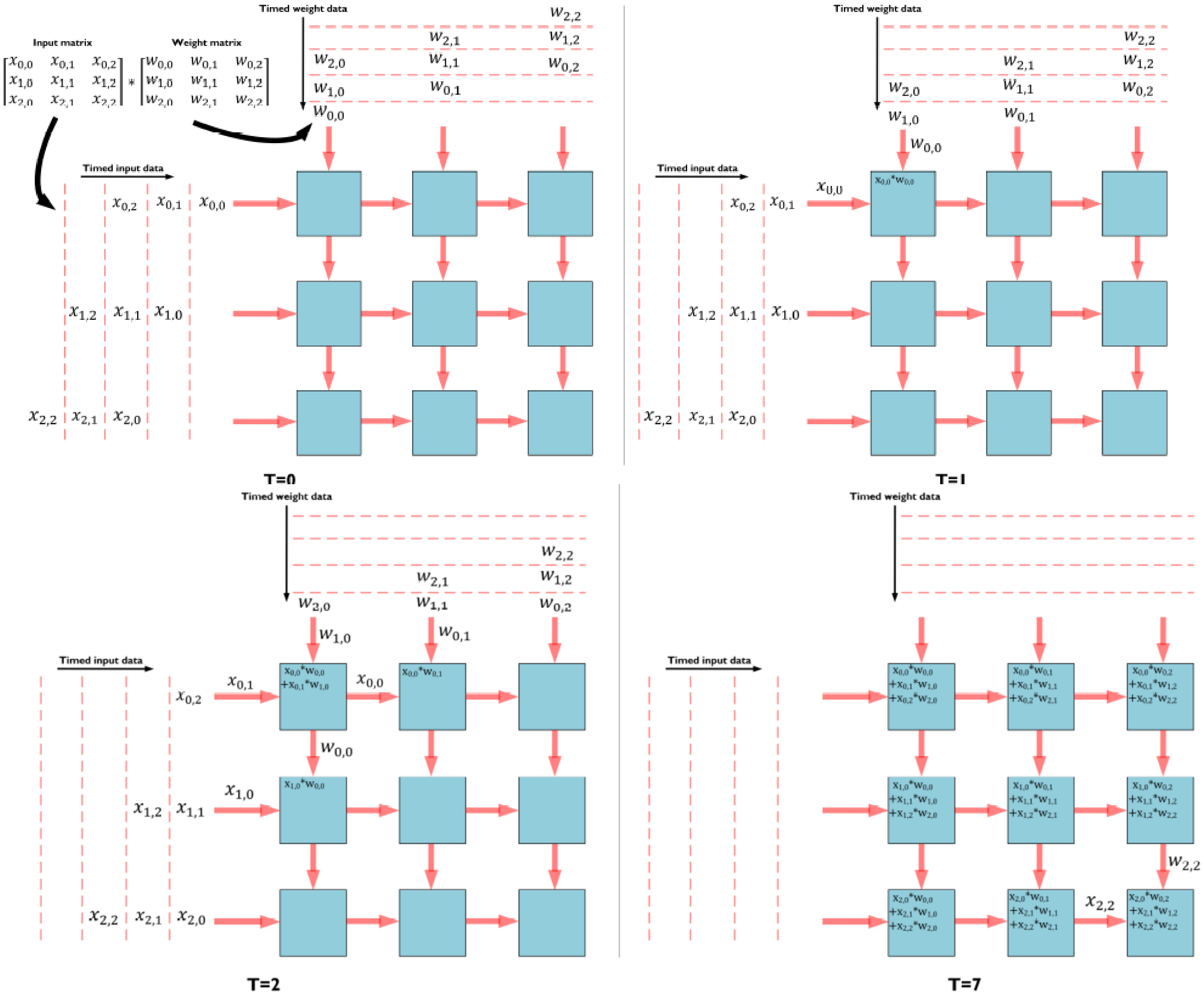
- Right Atrium
- Tricuspid Valve
- Right Ventricle
- Pulmonic Valve
- Pulmonary Arteries
- Pulmonic Veins
- Left Atrium
- Mitral Valve
- Left Ventricle
- Aortic Valve
- Aorta

Similar to blood flow: heart -> many cells -> heart
Memory: heart
Data: blood
PEs: cells

H.T. Kung, "Why systolic architectures?" IEEE Computer 1982)

Source: In-datacenter Performance Analysis of a Tensor Processing Unit Jouppi et al, ISCA, 6/2017

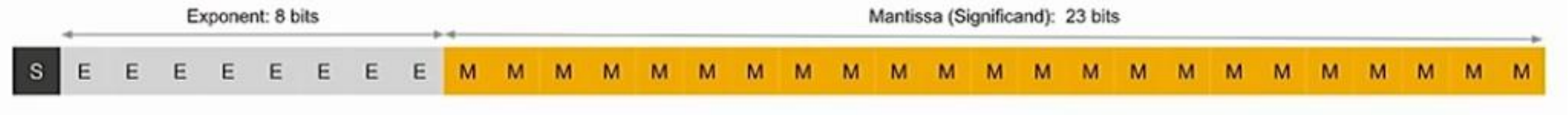
Systolic arrays for DNN acceleration (Ex. TPU)


Input matrix
Weight matrix
Timed weight data
Timed input data
T=0
T=1
T=2
T=7
x0,0*w0,0
x0,0*w0,0 +x0,1*w1,0
x0,0*w0,1
x1,0*w0,0
x0,0*w0,0 +x0,1*w1,0 +x0,2*w2,0
x0,0*w0,1 +x0,1*w1,1 +x0,2*w2,1
x0,0*w0,2 +x0,1*w1,2 +x0,2*w2,2
x1,0*w0,0 +x1,1*w1,0 +x1,2*w2,0
x1,0*w0,1 +x1,1*w1,1 +x1,2*w2,1
x1,0*w0,2 +x1,1*w1,2 +x1,2*w2,2
x2,0*w0,0 +x2,1*w1,0 +x2,2*w2,0
x2,0*w0,1 +x2,1*w1,1 +x2,2*w2,1
x2,0*w0,2 +x2,1*w1,2 +x2,2*w2,2

Ref. Azghadi2020
IEEE TRABS ON
BIOMEDICAL
CIRCUITS AND
SYSTEMS

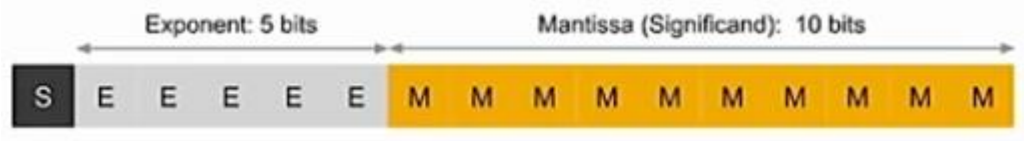
NN Training Works with Low-precision FP

fp32: Single-precision IEEE Floating Point Format


Exponent: 8 bits
Mantissa (Significand): 23 bits
S E E E E E E E E M M M M M M M M M M M M M M M M M M M M M M M

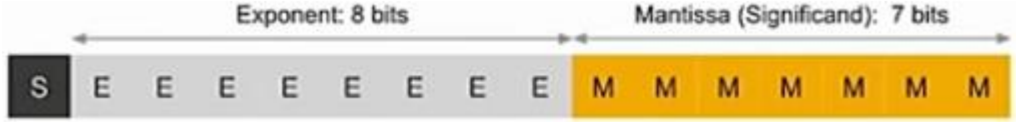
Range: (10^-45) to (10^38)

fp16: Half-precision IEEE Floating Point Format


Exponent: 5 bits
Mantissa (Significand): 10 bits
S E E E E E M M M M M M M M M M

Range: 10^-8 to 65504

bfloat16: Brain Floating Point Format


Exponent: 8 bits
Mantissa (Significand): 7 bits
S E E E E E E E E M M M M M M M

Range: (10^-45) to (10^38)

- ❖ Represent the same range of numbers of fp32 just at a much lower position.
- ❖ It turns out that we don’t need all that precision for NN training, but we do actually need all the range.

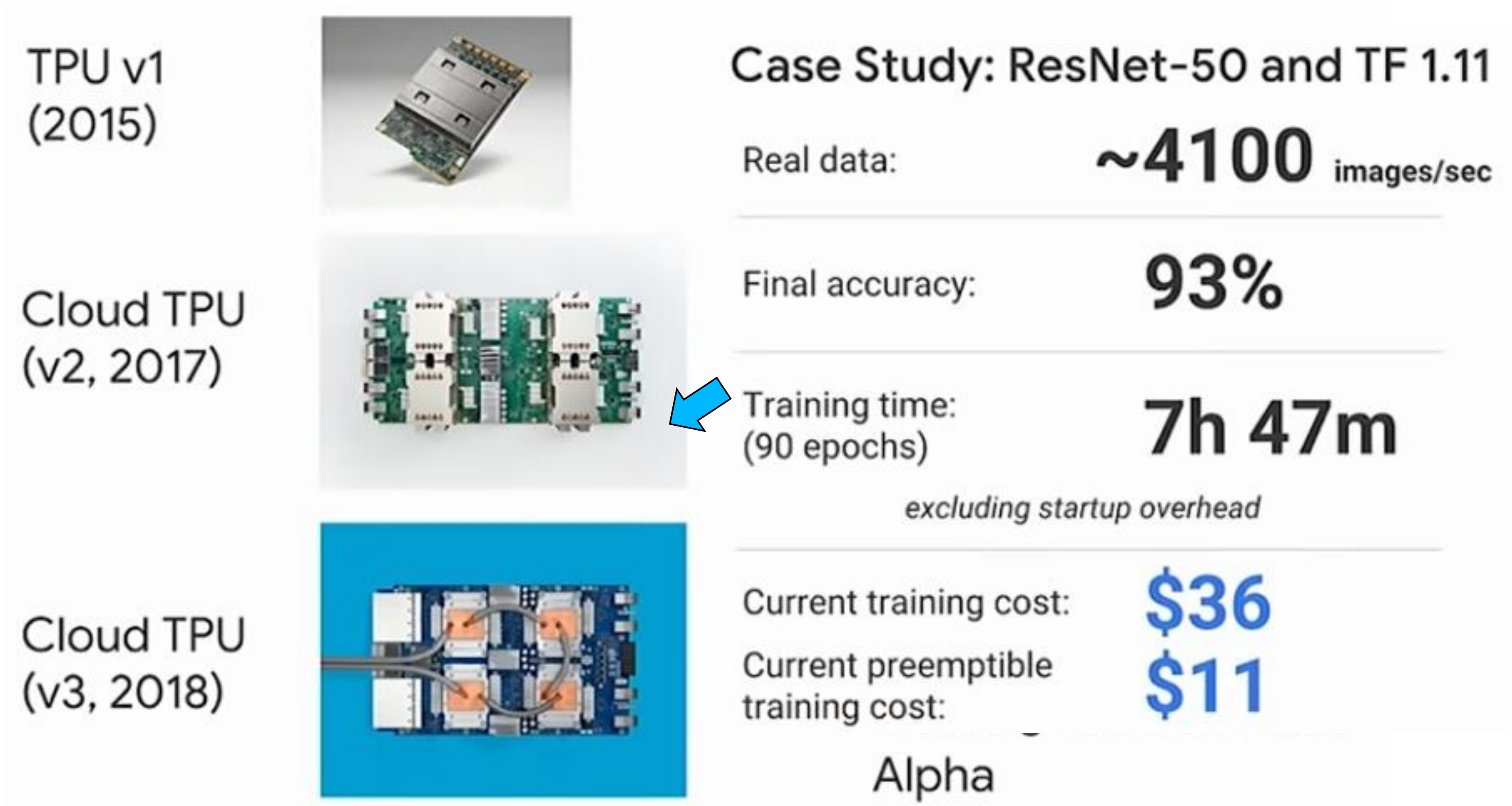
NN Training Works with Low-precision FP

- One technique exploited by the new chips is using **low-precision**, often fixed-point data, **eight bits** or even fewer, especially for inference.

- One of the major open questions in all of this as far as hardware accelerators are concerned is **how far can you actually push this down** without losing classification accuracy?

- Results from **Google, Intel, and others** show that such low-precision computations can be very powerful when the data is prepared correctly, which also opens opportunities for novel electronics.

What are the differences between the three TPUs



TPU v1 (2015)

Cloud TPU (v2, 2017)

Cloud TPU (v3, 2018)

Case Study: ResNet-50 and TF 1.11

Real data: ~4100 images/sec

Final accuracy: 93%

Training time: (90 epochs) 7h 47m

excluding startup overhead

Current training cost: $36

Current preemptible training cost: $11

Alpha

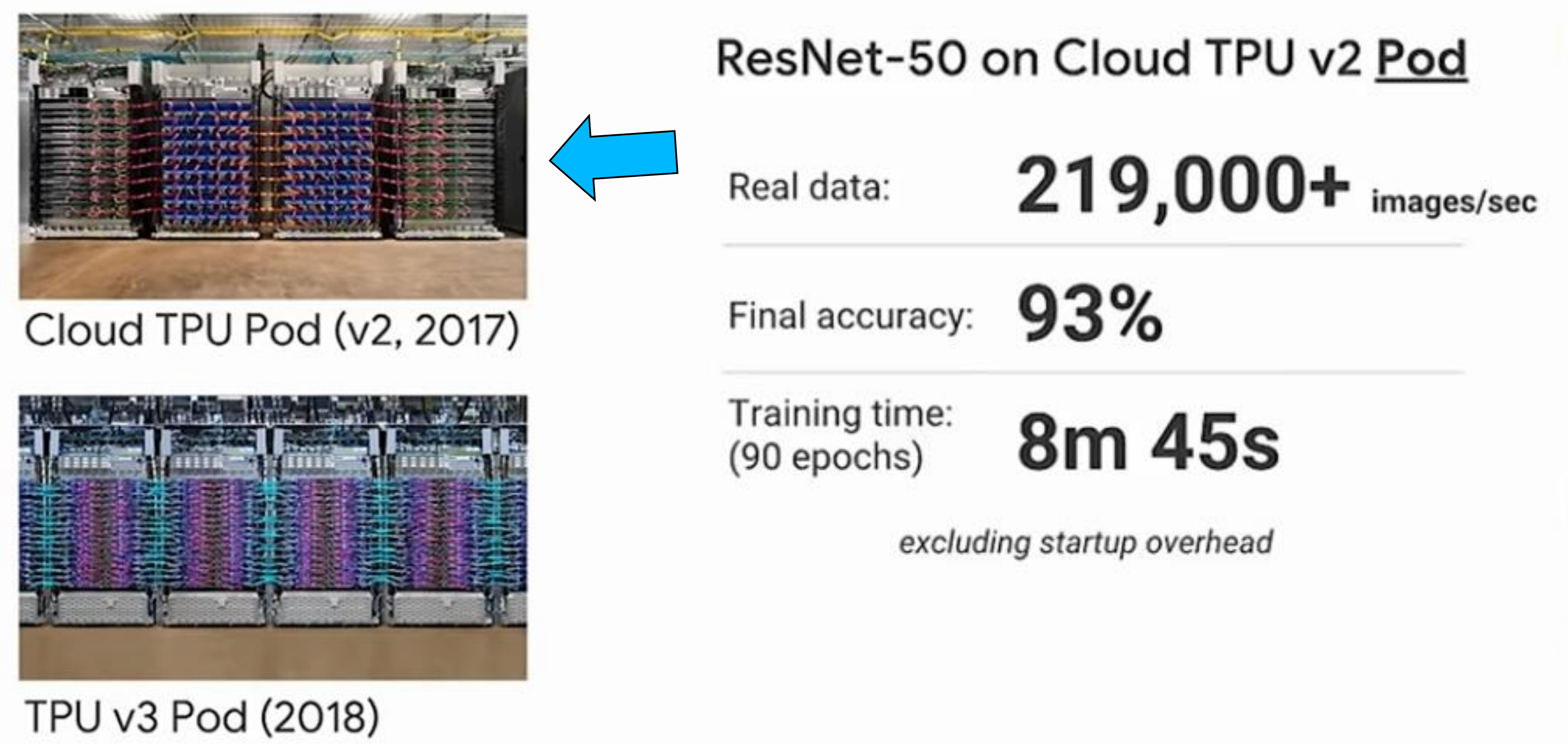
FLOPS -> OPS (Fixed-point operations per sec.)
e.g. PC(Core i7) ~500GFLOPS
Operation performance: TOPS/GOPS **(Tera/Giga Operations Per Second)**
Energy efficiency: TOPS/W **(Tera Ops. per sec. / Joule per sec = Tera ops. / Joule)**
Energy consumption per op. **(1/(TOPS/W) [pJ/op] = 1 [pJ/op])**

Synapse op. in **brain**: 0.1~1 fJ/op

(1fJ =10^{-15} joules)

1,000~10,000 TOPS/W
=1~10 POPS/W

What are the differences between the three TPUs



Cloud TPU Pod (v2, 2017)

TPU v3 Pod (2018)

ResNet-50 on Cloud TPU v2 Pod

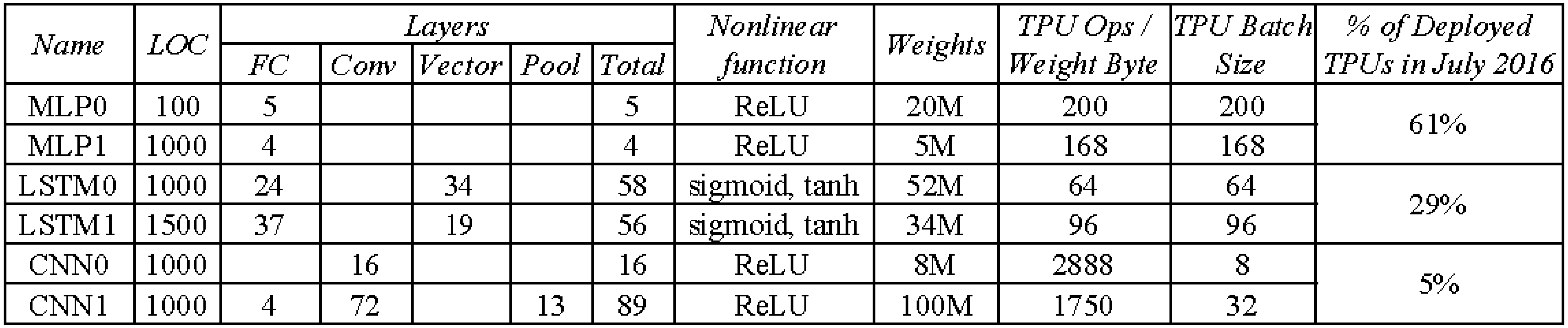
Real data:	219,000+ images/sec
Final accuracy:	93%
Training time: (90 epochs)	8m 45s

excluding startup overhead

https://cloud.google.com/tpu/

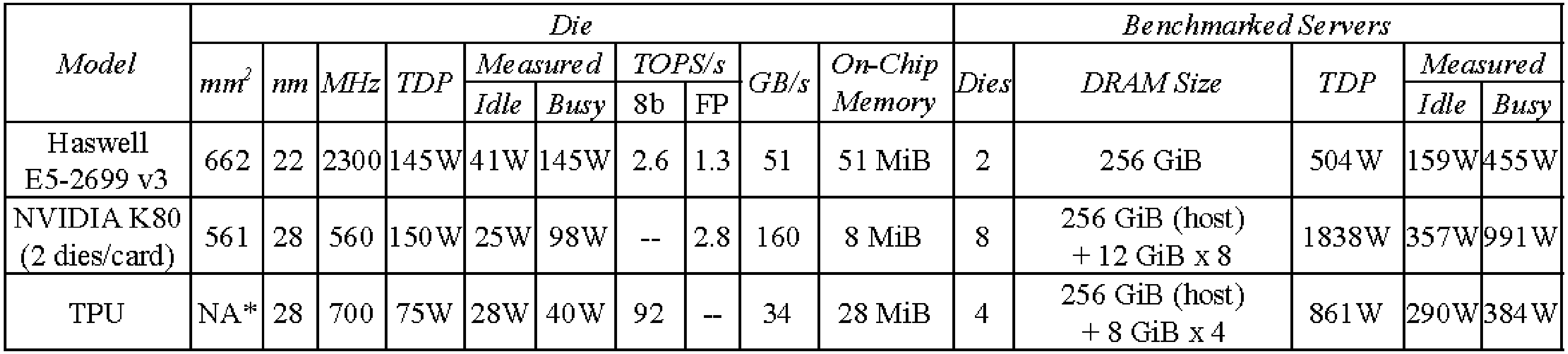
TPU Performance on three Popular NNs

- Multi-Layer Perceptrons (MLP)
- Convolutional Neural Networks (CNN)
- Recurrent Neural Networks (RNN)



Name	*LOC*	*Layers*					*Nonlinear function*	*Weights*	*TPU Ops / Weight Byte*	*TPU Batch Size*	*% of Deployed TPUs in July 2016*
		FC	*Conv*	*Vector*	*Pool*	*Total*					
MLP0	100	5				5	ReLU	20M	200	200	61%
MLP1	1000	4				4	ReLU	5M	168	168	
LSTM0	1000	24		34		58	sigmoid, tanh	52M	64	64	29%
LSTM1	1500	37		19		56	sigmoid, tanh	34M	96	96	
CNN0	1000		16			16	ReLU	8M	2888	8	5%
CNN1	1000	4	72		13	89	ReLU	100M	1750	32	

Tensor Processing Unit (TPU) with MLP, CNN, RNN)

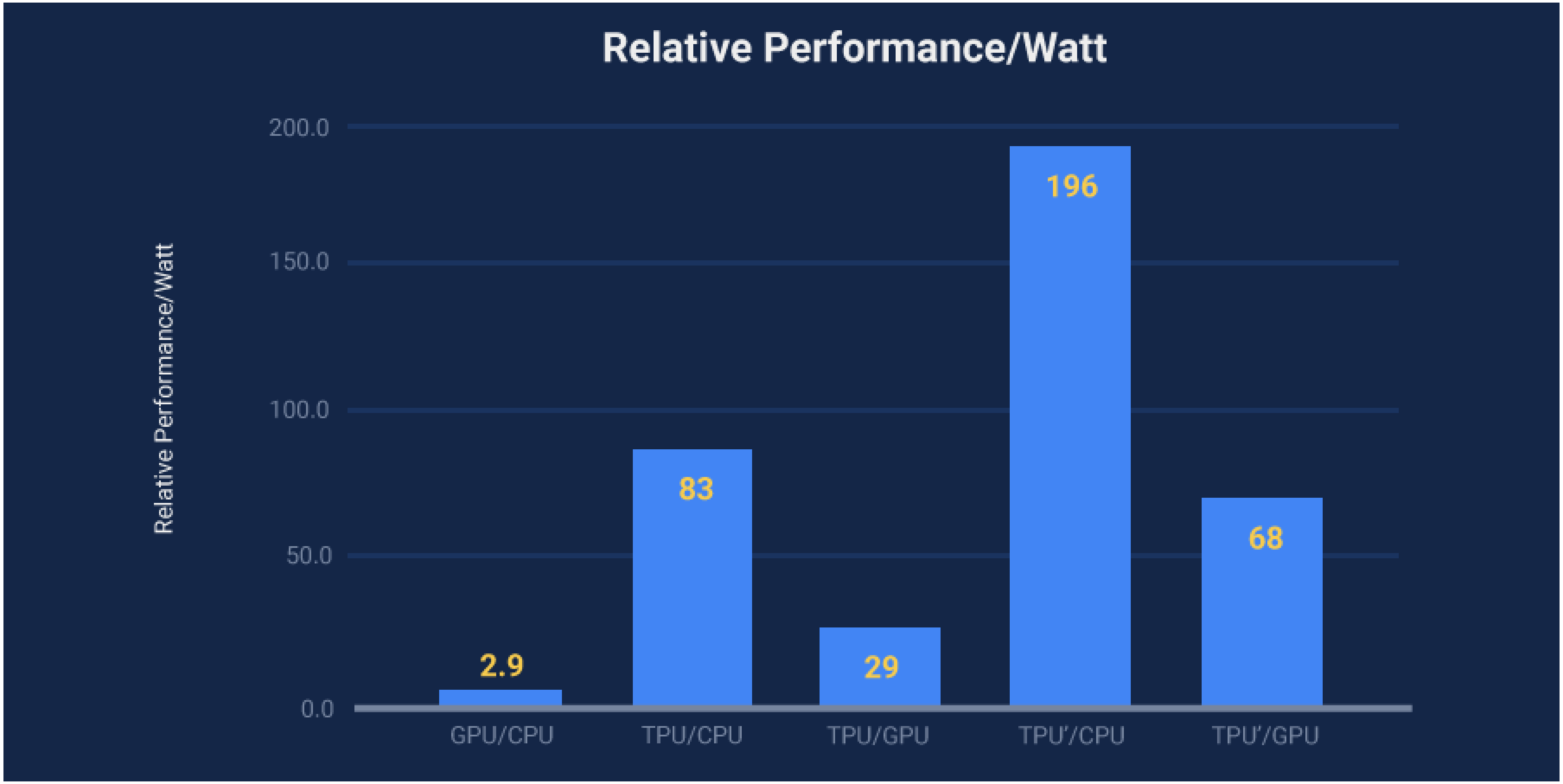


Model	*Die*										*Benchmarked Servers*				
	mm^2	*nm*	*MHz*	*TDP*	*Measured*		*TOPS/s*		*GB/s*	*On-Chip Memory*	*Dies*	*DRAM Size*	*TDP*	*Measured*	
					Idle	*Busy*	8b	FP						*Idle*	*Busy*
Haswell E5-2699 v3	662	22	2300	145W	41W	145W	2.6	1.3	51	51 MiB	2	256 GiB	504W	159W	455W
NVIDIA K80 (2 dies/card)	561	28	560	150W	25W	98W	--	2.8	160	8 MiB	8	256 GiB (host) + 12 GiB x 8	1838W	357W	991W
TPU	NA*	28	700	75W	28W	40W	92	--	34	28 MiB	4	256 GiB (host) + 8 GiB x 4	861W	290W	384W

Benchmarked servers use Haswell CPUs, K80 GPUs, and TPUs. Haswell has 18 cores, and the K80 has 13 SMX processors.

Source: In-datacenter Performance Analysis of a Tensor Processing Unit, Jouppi et al, ISCA, 6/2017

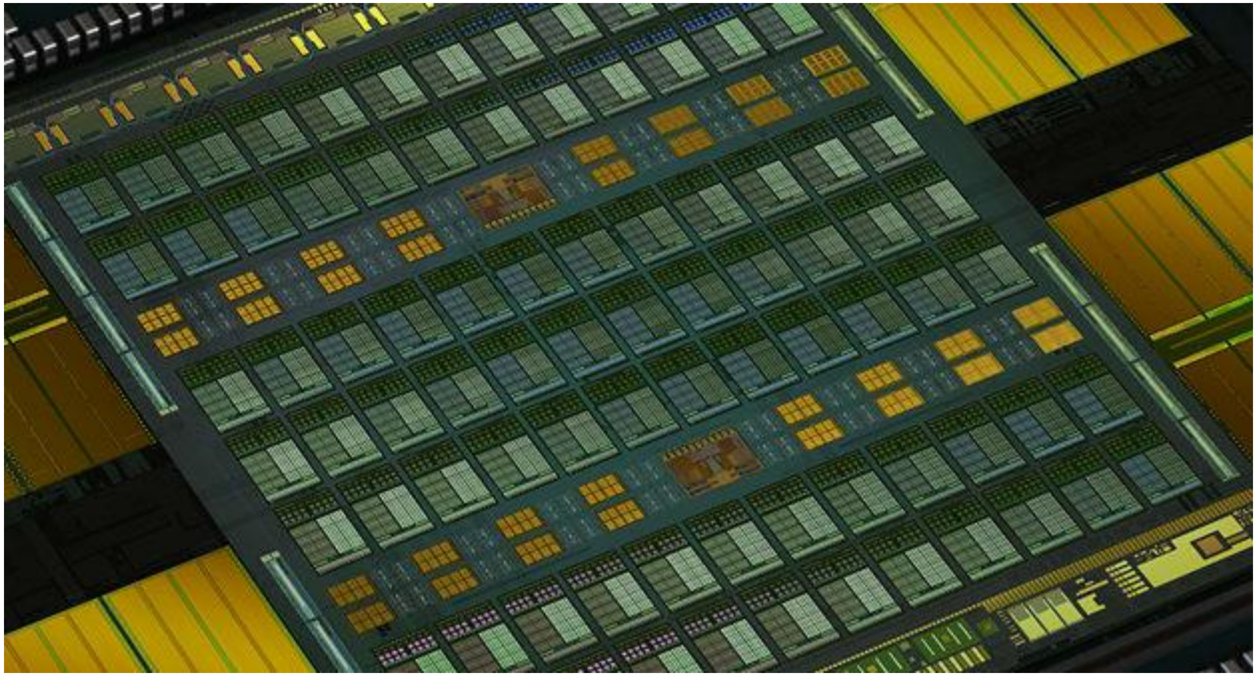
TPU Relative Performance/Watt



Quantifying the performance of the TPU, our first machine learning chip:
https://cloudplatform.googleblog.com/2017/04/quantifying-the-performance-of-the-TPU-our-first-machine-learning-chip.html

NVIDIA's Volta GPU is Specially Designed for AI

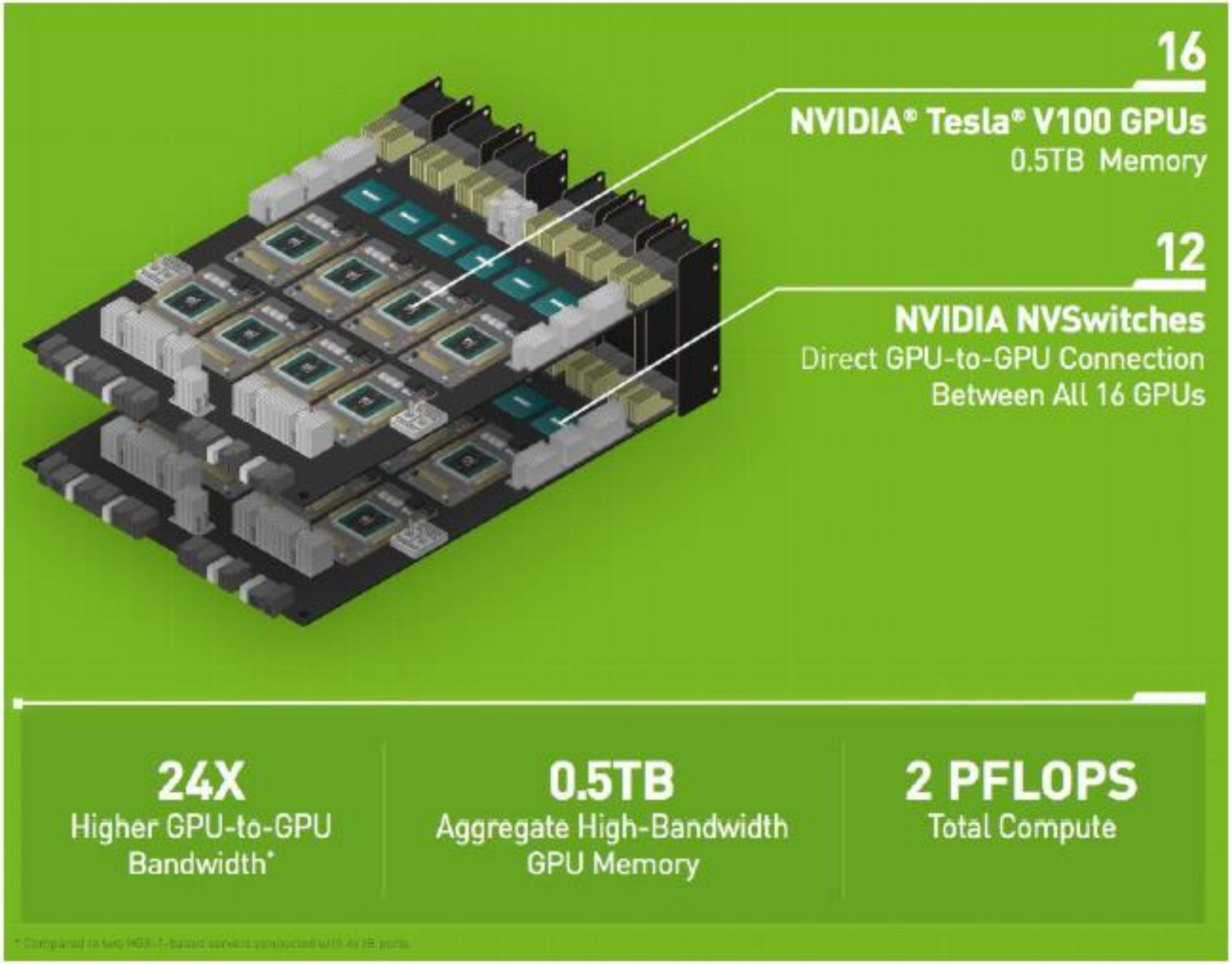
- NVIDIA's Volta GPU is specially designed for ML, and it offers 100 TFLOPS of DL performance, according to the company.
- GPUs were built for graphics workloads and *evolved* for high performance computing and AI workloads
- While GPUs are used extensively for training, they're not really needed for inference



NVIDIA's Volta GPU architecture is specially designed for AI. (Image courtesy of NVIDIA.)

The HGX-2, announced at NVIDIA GTC May 2018

Multi-precision computing platform for scientific computing (high precision) and AI workloads (low precision).



16
NVIDIA® Tesla® V100 GPUs
0.5TB Memory

12
NVIDIA NVSwitches
Direct GPU-to-GPU Connection
Between All 16 GPUs

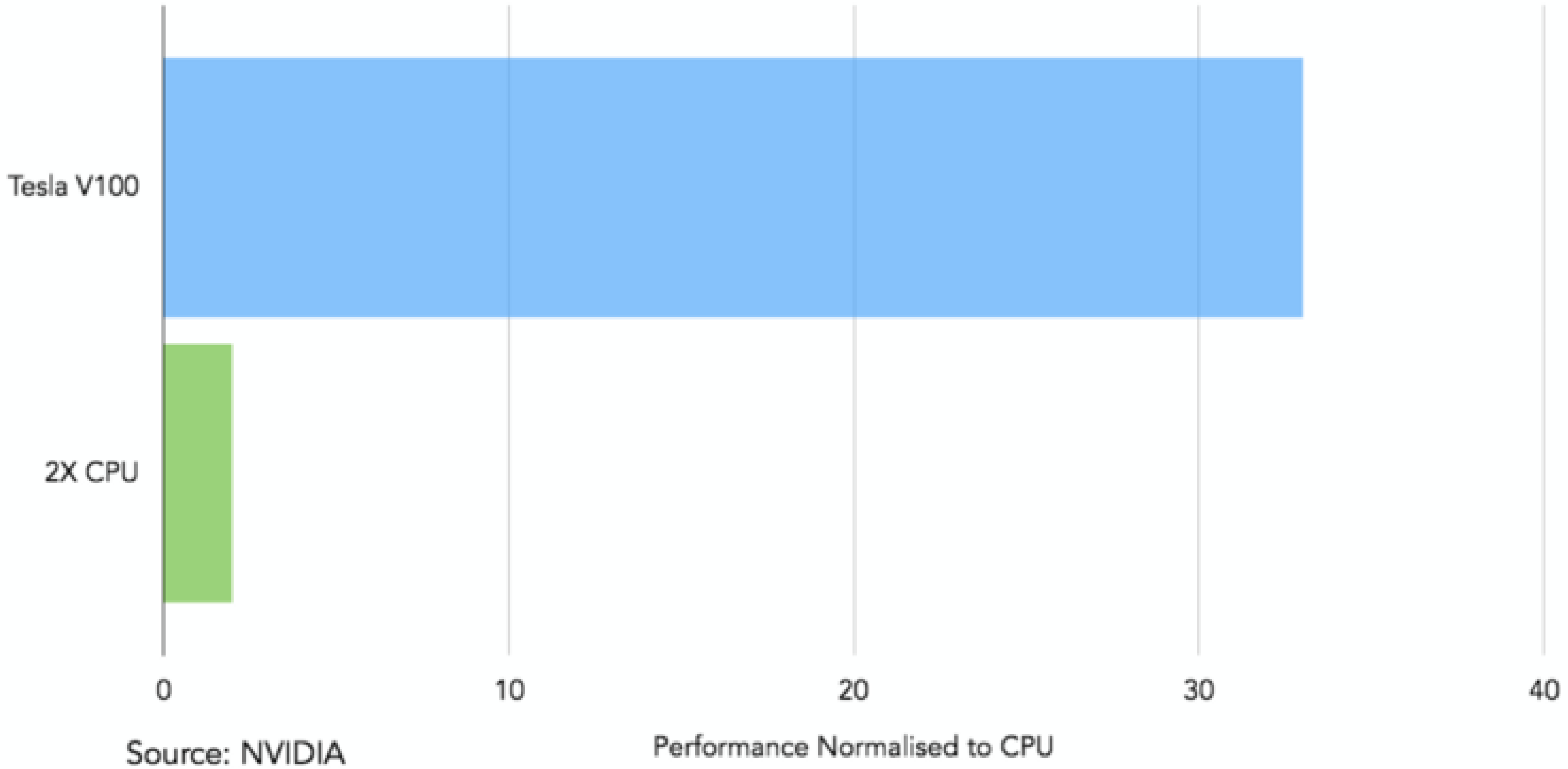
24X
Higher GPU-to-GPU Bandwidth*

0.5TB
Aggregate High-Bandwidth GPU Memory

2 PFLOPS
Total Compute

* Compared to two HGX-1-based servers connected with 4x IB ports.

NVIDIA’s GPU Performance

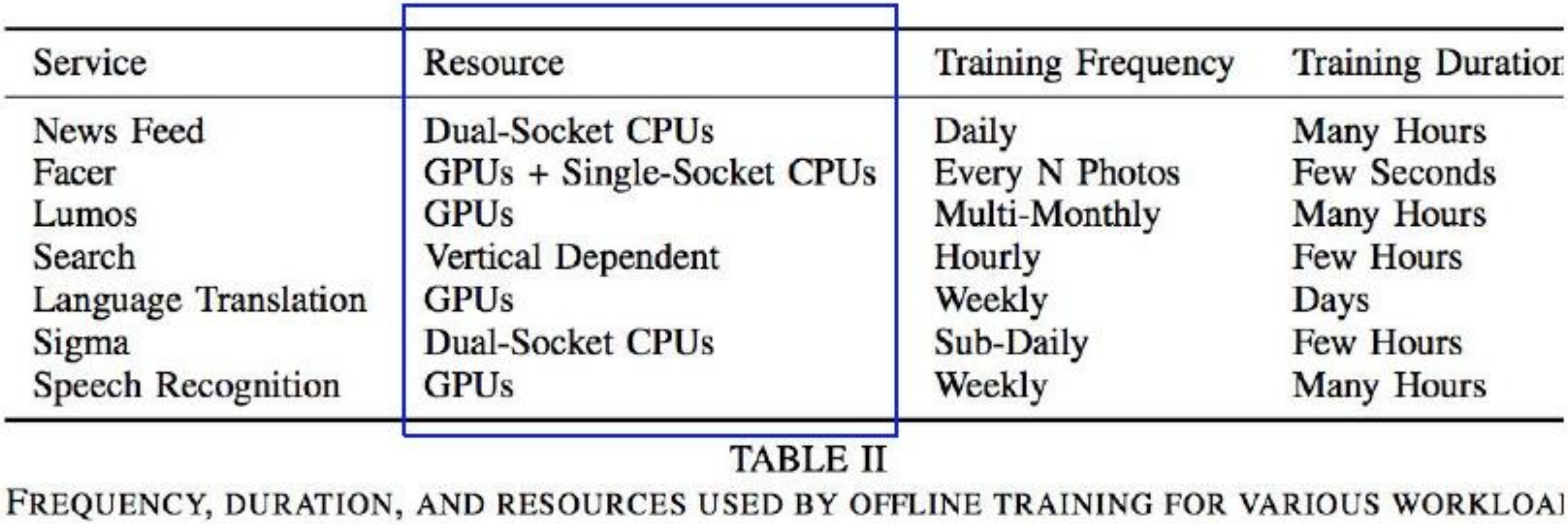


30x Higher Throughput than CPU Server on Deep Learning Inference

Source: NVIDIA

At Facebook, for example, primary use case of GPUs is offline training rather than serving real-time data to users

Offline training uses a mix of GPUs and CPUs

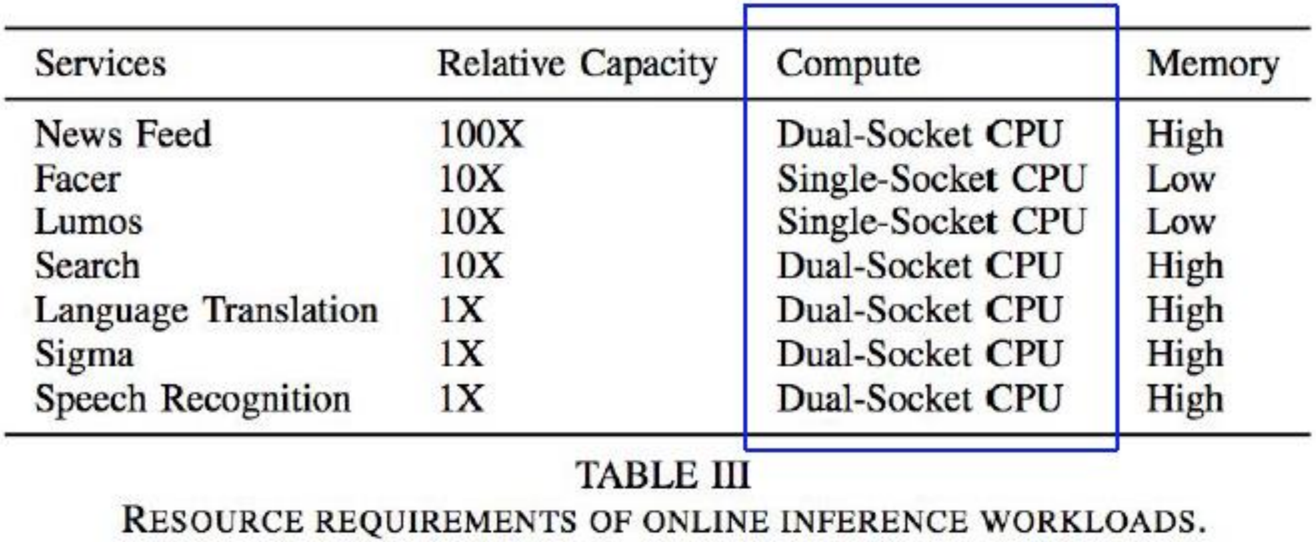


Service	Resource	Training Frequency	Training Duratior
News Feed	Dual-Socket CPUs	Daily	Many Hours
Facer	GPUs + Single-Socket CPUs	Every N Photos	Few Seconds
Lumos	GPUs	Multi-Monthly	Many Hours
Search	Vertical Dependent	Hourly	Few Hours
Language Translation	GPUs	Weekly	Days
Sigma	Dual-Socket CPUs	Sub-Daily	Few Hours
Speech Recognition	GPUs	Weekly	Many Hours

TABLE II

FREQUENCY, DURATION, AND RESOURCES USED BY OFFLINE TRAINING FOR VARIOUS WORKLOAI

However, online training is CPU-heavy



Services	Relative Capacity	Compute	Memory
News Feed	100X	Dual-Socket CPU	High
Facer	10X	Single-Socket CPU	Low
Lumos	10X	Single-Socket CPU	Low
Search	10X	Dual-Socket CPU	High
Language Translation	1X	Dual-Socket CPU	High
Sigma	1X	Dual-Socket CPU	High
Speech Recognition	1X	Dual-Socket CPU	High

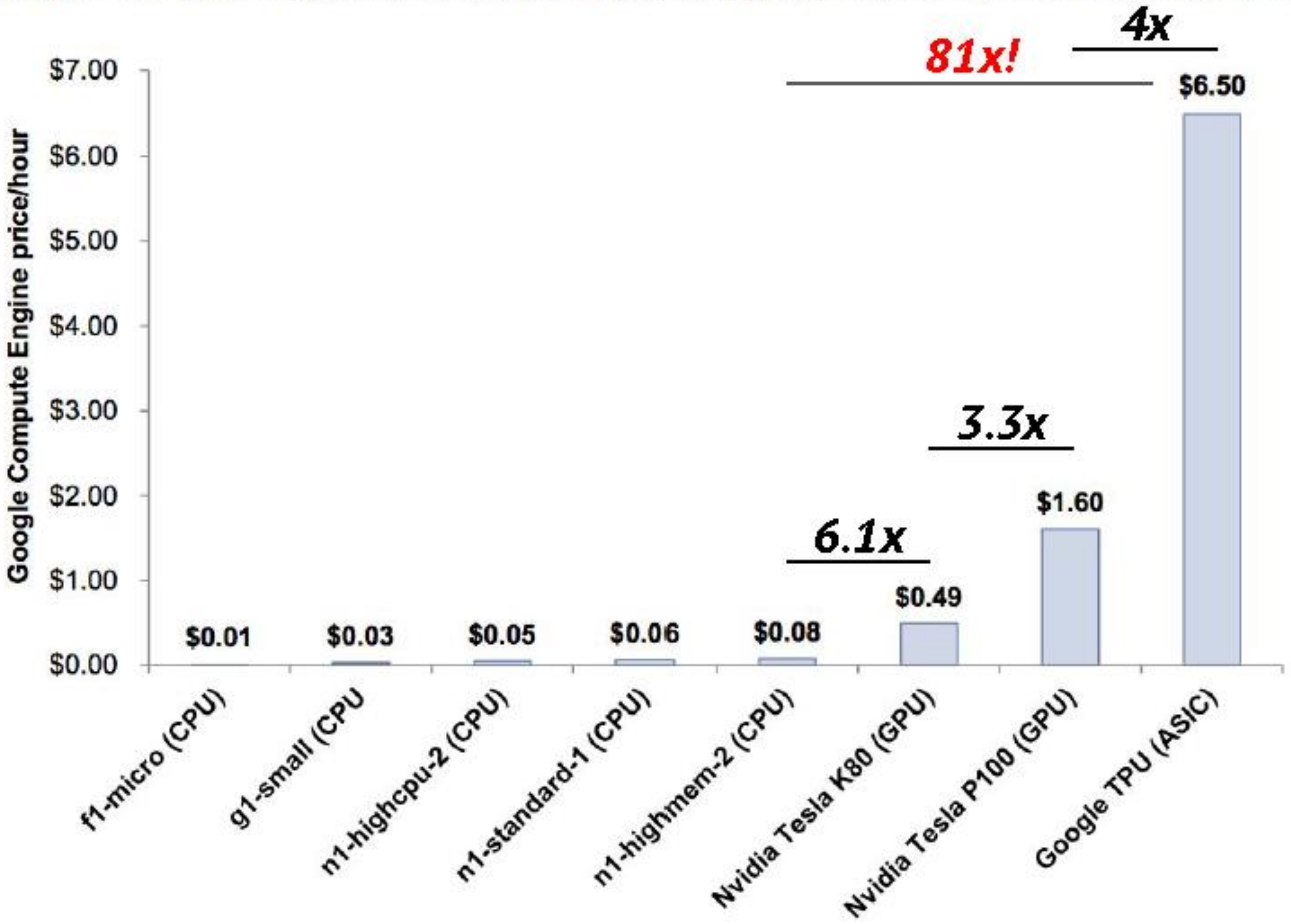
TABLE III

RESOURCE REQUIREMENTS OF ONLINE INFERENCE WORKLOADS.

GPUs & ASICs Renting Cost Per Hour

Exhibit 11: Offerings for AI command significantly higher prices

Google Compute Engine price/hour/single compute instance (i.e. per 1CPU, GPU, TPU, etc)

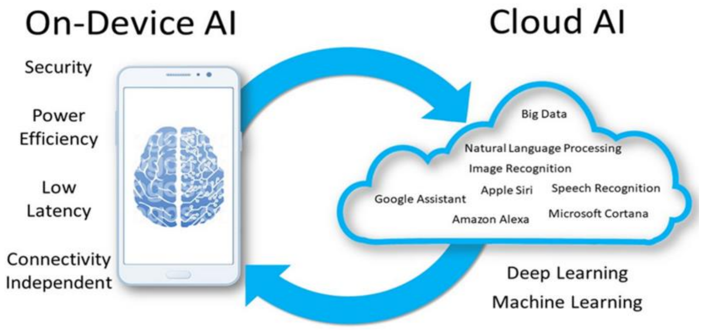


Source: Google, Goldman Sachs Global Investment Research.

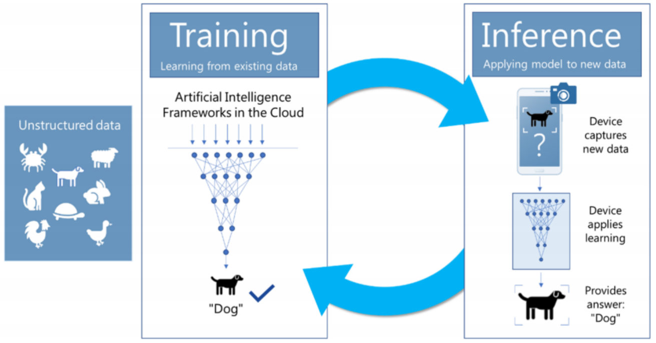
Mobile AI-Chips

Mobile AI-Chips

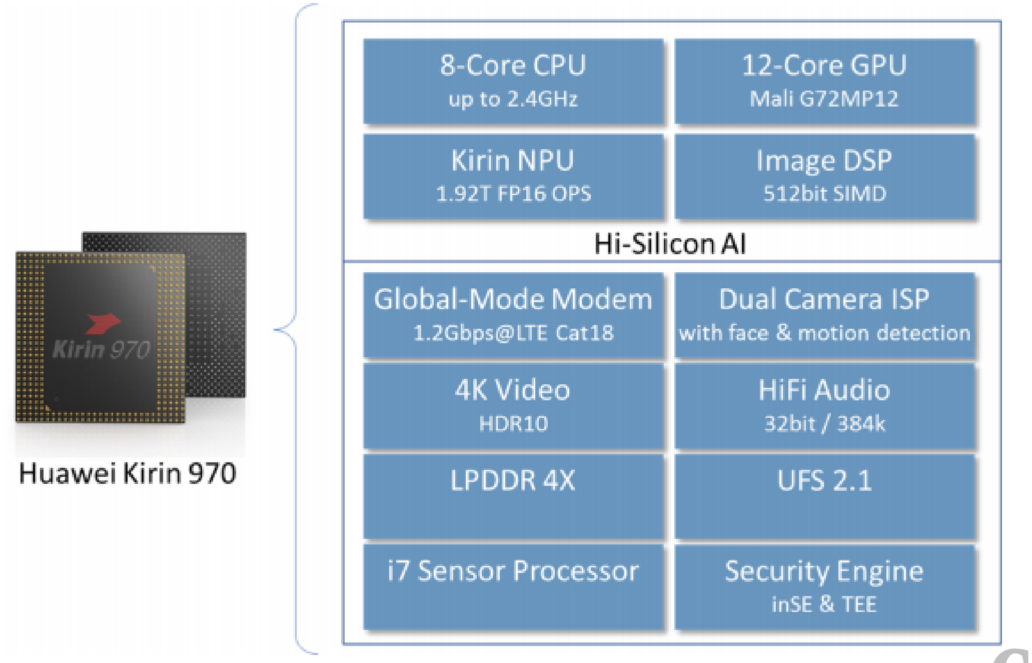
- Much of the data captured by the smartphone, including images, video, and sound, is unstructured.
- **Training and Inference are Two Vital Components of AI on Smartphones**.
- Unlike structured data — information with a degree of organization — unstructured data makes compilation a time- and energy-consuming task.
- Huawei's Kirin 970 chipset comes with its own **neural processing unit (NPU).**
- Huawei has it own APIs that developers need to use to tap the power of the "neural" hardware.
- Google has it mobile AI framework - TensorFlow Lite.



Source: IDC, Huawei, Qualcomm, 2017

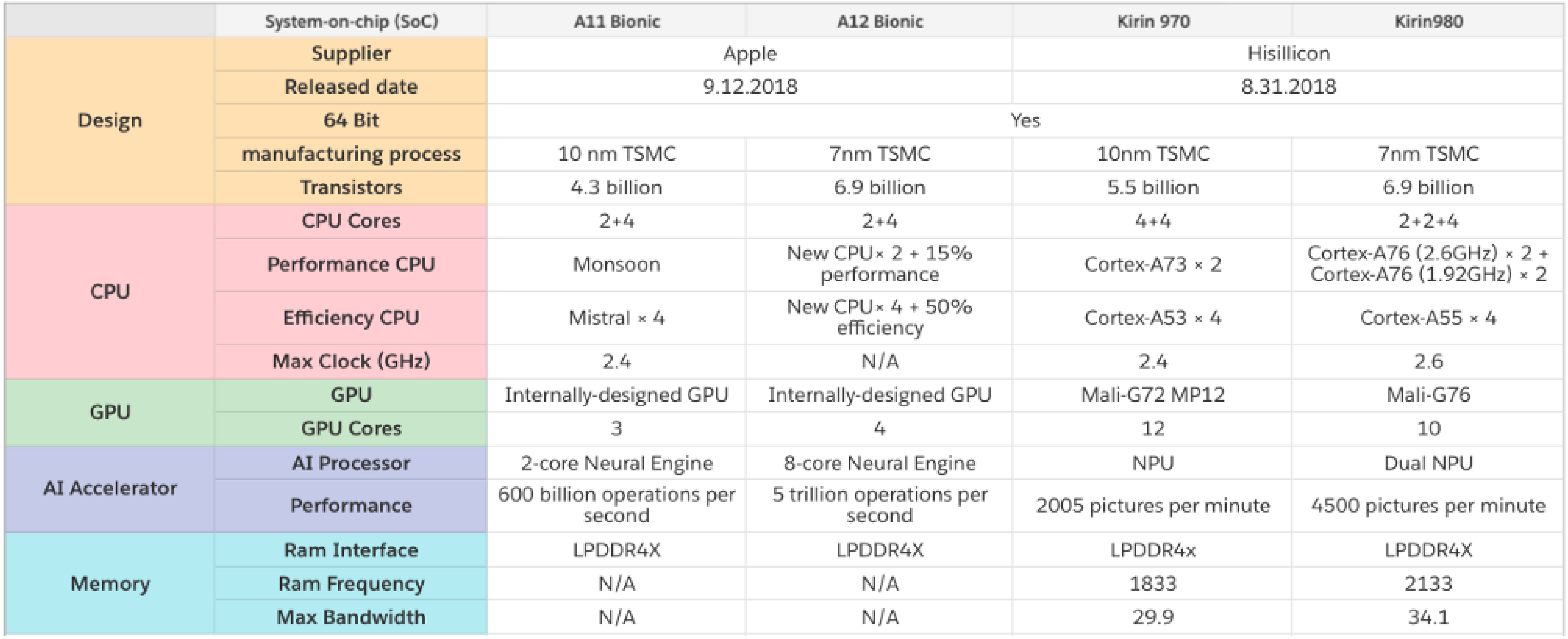


Source: IDC, Huawei, Qualcomm, Nvidia, 2017



Source: Huawei, 2017

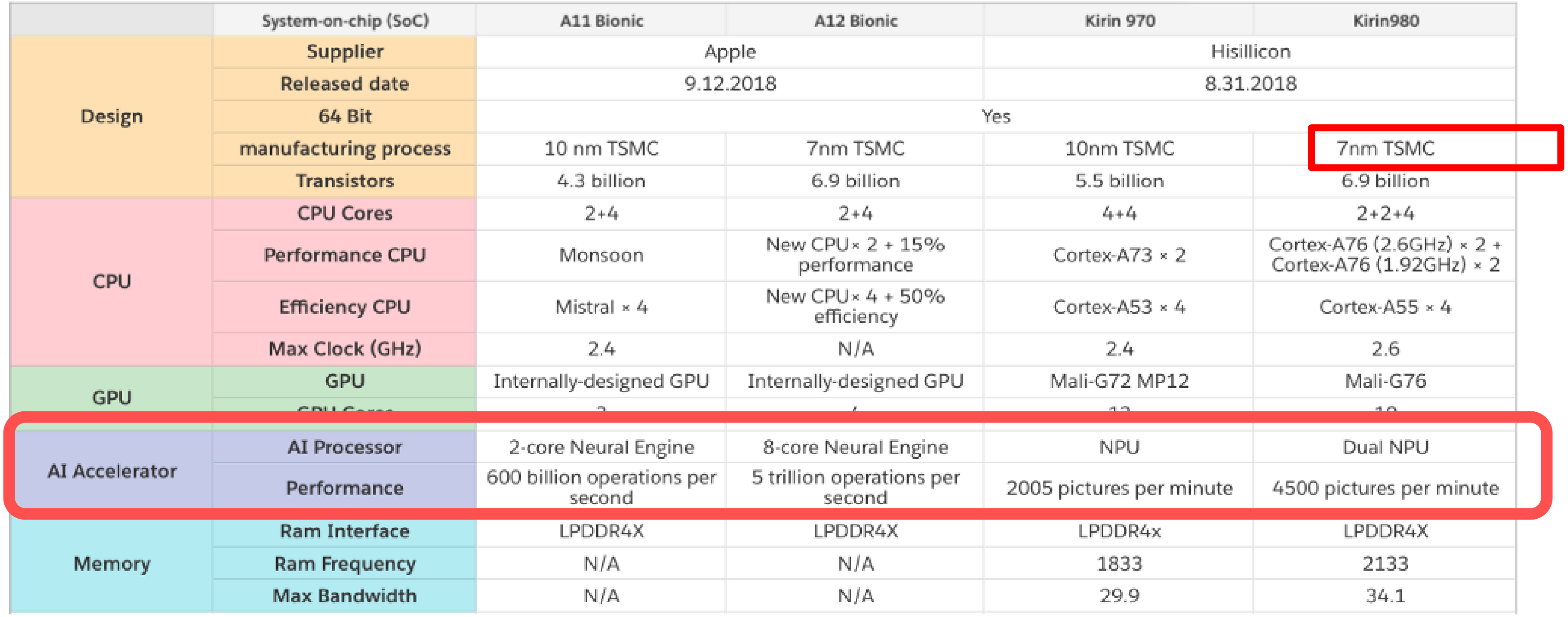
Summary of Mobile AI Chips



	System-on-chip (SoC)	A11 Bionic	A12 Bionic	Kirin 970	Kirin980
Design	Supplier	Apple		Hisillicon	
	Released date	9.12.2018		8.31.2018	
	64 Bit	Yes			
	manufacturing process	10 nm TSMC	7nm TSMC	10nm TSMC	7nm TSMC
	Transistors	4.3 billion	6.9 billion	5.5 billion	6.9 billion
CPU	CPU Cores	2+4	2+4	4+4	2+2+4
	Performance CPU	Monsoon	New CPU× 2 + 15% performance	Cortex-A73 × 2	Cortex-A76 (2.6GHz) × 2 + Cortex-A76 (1.92GHz) × 2
	Efficiency CPU	Mistral × 4	New CPU× 4 + 50% efficiency	Cortex-A53 × 4	Cortex-A55 × 4
	Max Clock (GHz)	2.4	N/A	2.4	2.6
GPU	GPU	Internally-designed GPU	Internally-designed GPU	Mali-G72 MP12	Mali-G76
	GPU Cores	3	4	12	10
AI Accelerator	AI Processor	2-core Neural Engine	8-core Neural Engine	NPU	Dual NPU
	Performance	600 billion operations per second	5 trillion operations per second	2005 pictures per minute	4500 pictures per minute
Memory	Ram Interface	LPDDR4X	LPDDR4X	LPDDR4x	LPDDR4X
	Ram Frequency	N/A	N/A	1833	2133
	Max Bandwidth	N/A	N/A	29.9	34.1

Source: medium.com

Summary of Mobile AI Chips



	System-on-chip (SoC)	A11 Bionic	A12 Bionic	Kirin 970	Kirin980
Design	Supplier	Apple		Hisillicon	
	Released date	9.12.2018		8.31.2018	
	64 Bit	Yes			
	manufacturing process	10 nm TSMC	7nm TSMC	10nm TSMC	7nm TSMC
	Transistors	4.3 billion	6.9 billion	5.5 billion	6.9 billion
CPU	CPU Cores	2+4	2+4	4+4	2+2+4
	Performance CPU	Monsoon	New CPU× 2 + 15% performance	Cortex-A73 × 2	Cortex-A76 (2.6GHz) × 2 + Cortex-A76 (1.92GHz) × 2
	Efficiency CPU	Mistral × 4	New CPU× 4 + 50% efficiency	Cortex-A53 × 4	Cortex-A55 × 4
	Max Clock (GHz)	2.4	N/A	2.4	2.6
GPU	GPU	Internally-designed GPU	Internally-designed GPU	Mali-G72 MP12	Mali-G76
	GPU Cores	3	4	12	10
AI Accelerator	AI Processor	2-core Neural Engine	8-core Neural Engine	NPU	Dual NPU
	Performance	600 billion operations per second	5 trillion operations per second	2005 pictures per minute	4500 pictures per minute
Memory	Ram Interface	LPDDR4X	LPDDR4X	LPDDR4x	LPDDR4X
	Ram Frequency	N/A	N/A	1833	2133
	Max Bandwidth	N/A	N/A	29.9	34.1

Source: medium.com

Edge and IoT AI-Chips

~Processing Real-Time Data~

Edge Computing: Edge AI Chip

- The need for no latency, higher security, faster computing, and less dependence on connectivity will drive the adoption of devices that

On-device approach helps reduce latency for critical applications, lower dependence on the cloud, and better manage the massive data being generated by the IoT device.

D

computing

+

intelligence

where it is needed.

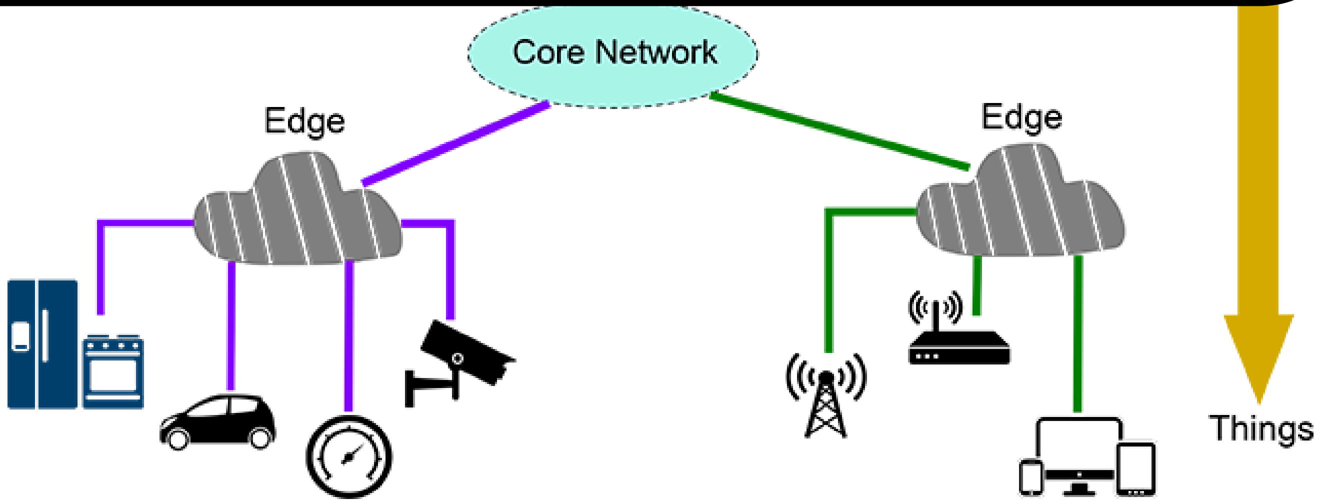
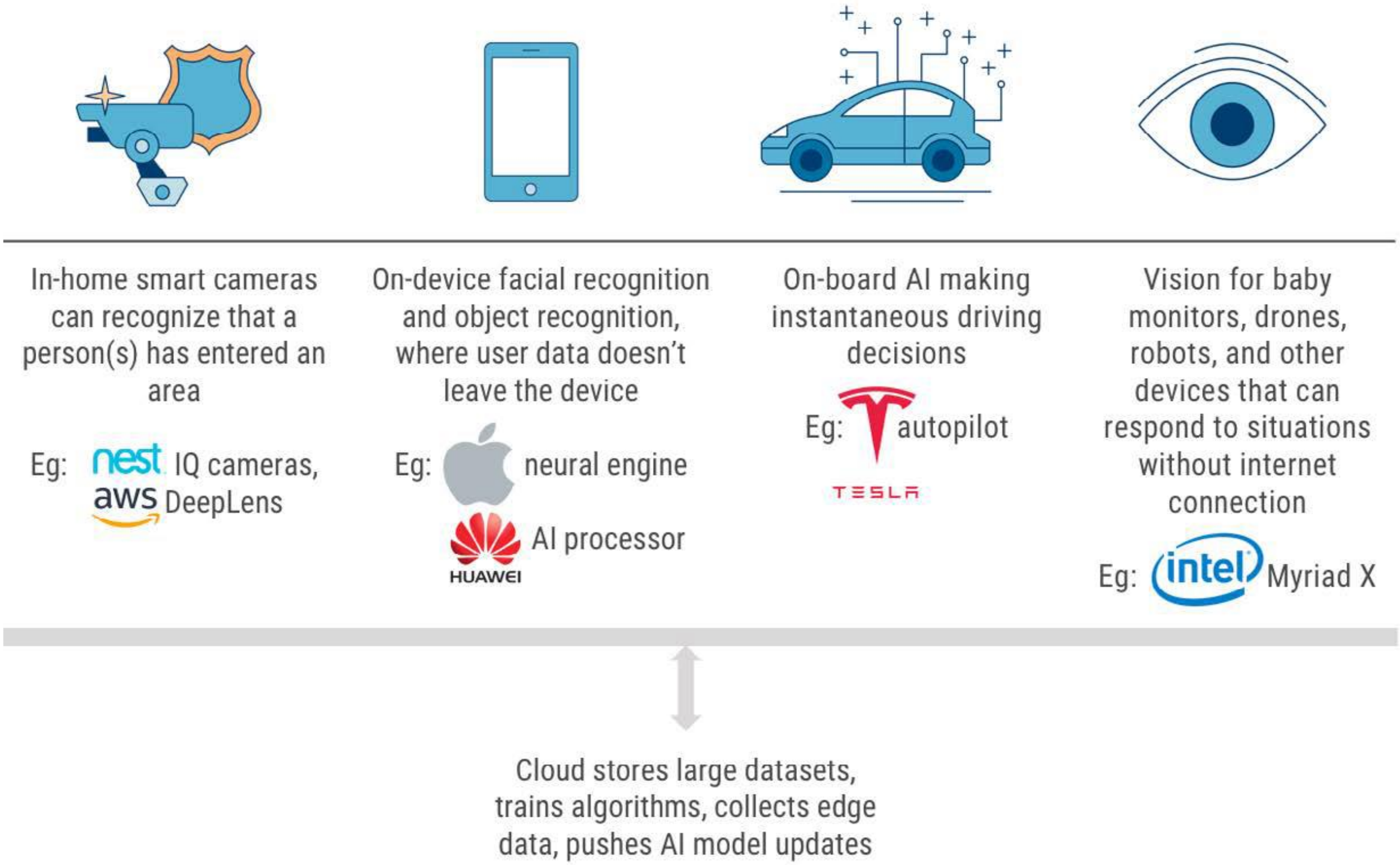


Illustration of an Edge Computing Architecture

Examples of Edge AI Applications



In-home smart cameras can recognize that a person(s) has entered an area

Eg: nest IQ cameras, aws DeepLens

On-device facial recognition and object recognition, where user data doesn't leave the device

Eg: neural engine

HUAWEI AI processor

On-board AI making instantaneous driving decisions

Eg: autopilot

TESLA

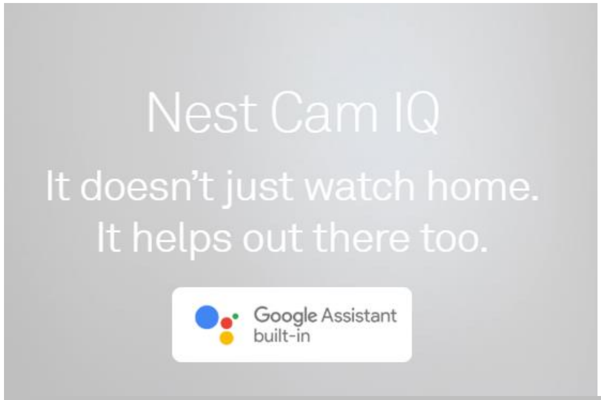
Vision for baby monitors, drones, robots, and other devices that can respond to situations without internet connection

Eg: intel Myriad X

Cloud stores large datasets, trains algorithms, collects edge data, pushes AI model updates

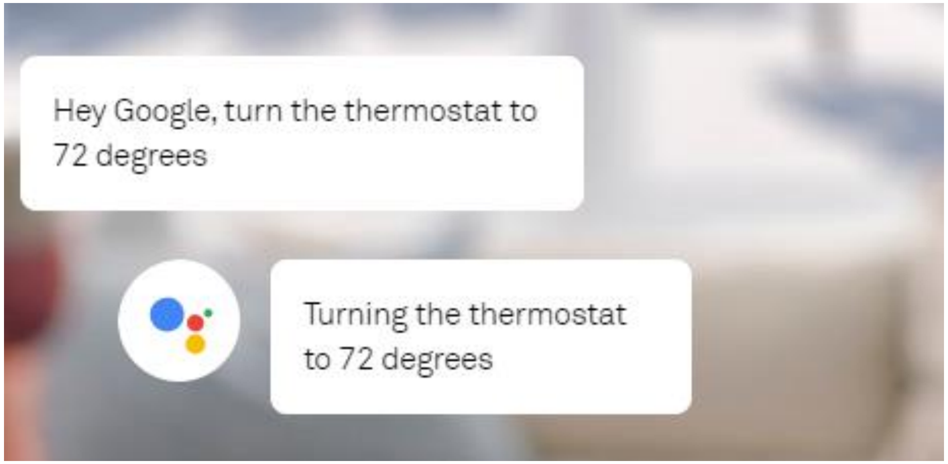
Source: CBINSIGHTS 2018

Examples of Edge AI Applications



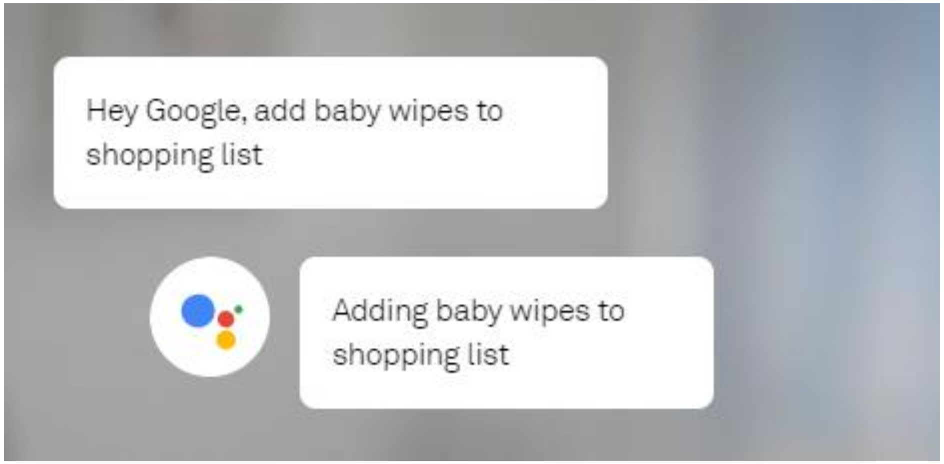
Combining a 4K sensor with HDR and Intelligent Imaging

Uses on-device vision processing to watch for motion, distinguish family members, and send alerts only if someone is not recognized or doesn't fit pre-defined parameters.



Hey Google, turn the thermostat to 72 degrees

Turning the thermostat to 72 degrees



Hey Google, add baby wipes to shopping list

Adding baby wipes to shopping list

https://nest.com/cameras/nest-cam-iq-indoor/overview/

Apple, Intel, and Google Edge AI-Chips

- **Apple** released its A11 chip with a "neural engine" for iPhone 8 and X. Apple claims it can perform machine learning tasks at up to 600B operations per second.
  - It powers new iPhone features like FaceID, which scans a user's face with an invisible spray of light, without uploading or storing any user data (or their face) in the cloud.

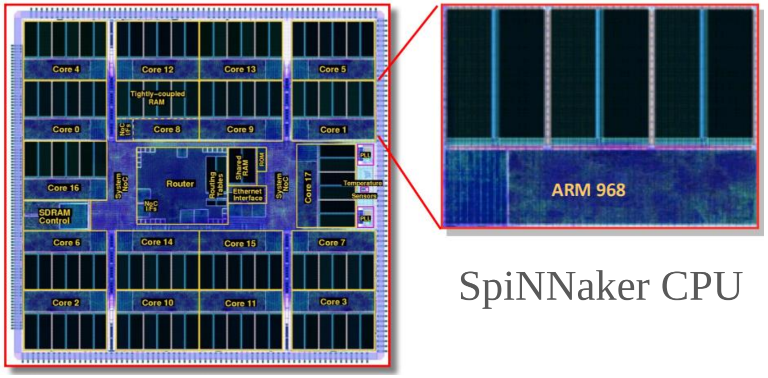
- **Intel** released an on-device vision processing chip called Myriad X (initially developed by Movidius, which Intel acquired in 2016).
  - Myriad X promises to take on-device deep learning beyond smartphones to devices like baby monitors and drones

- **Google** proposed a similar concept with its "federated learning" approach, where some of the machine learning "training" can happen on your device. It's testing out the feature in Gboard, the Google keyboard.

❖ AI on the edge reduces latency. But unlike the cloud, edge has storage and processing constraints.

Healthcare AI-Chips

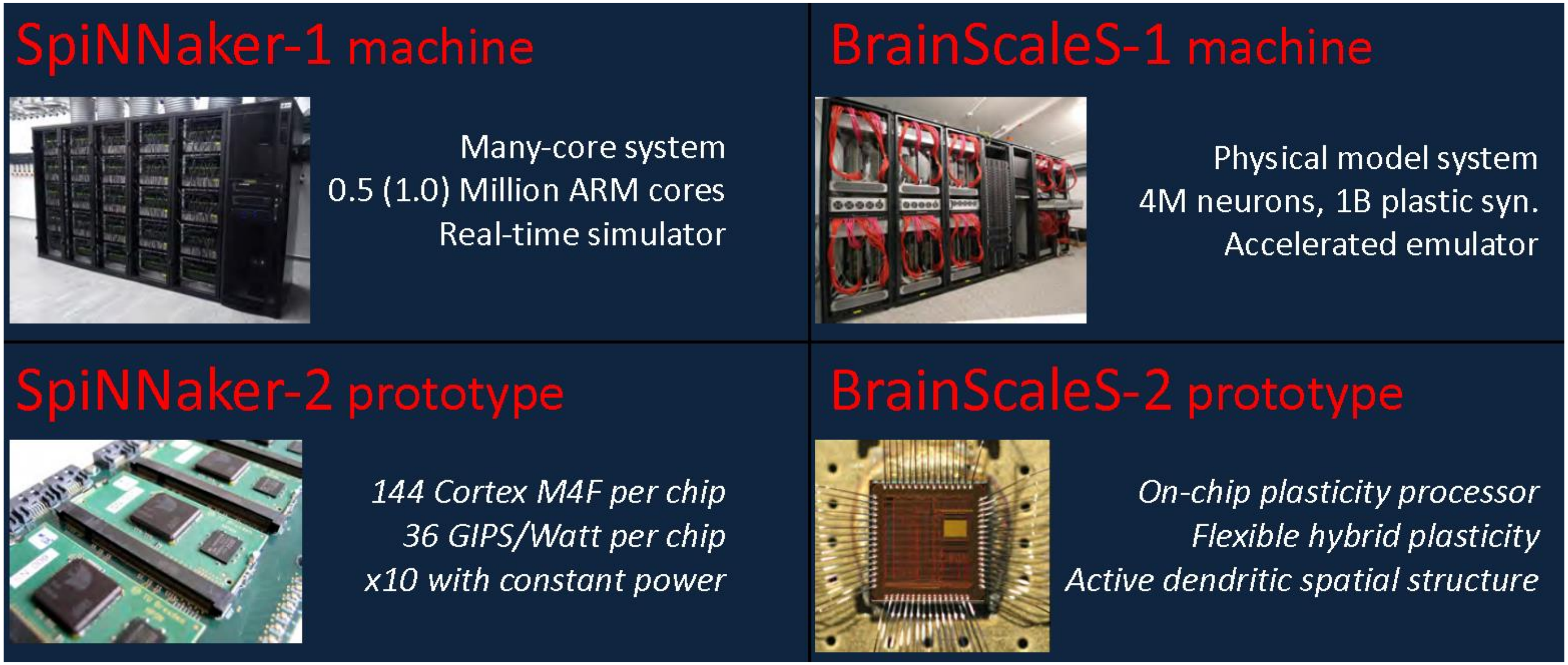
Healthcare AI-Chips



SpiNNaker CPU

Applications/Research Areas

- **Neuroscience**: neuroinformatics; brain simulation
- **Medicine**: medical informatics; early diagnosis; personalized treatment
- **Future computing**: interactive supercomputing; neuromorphic computing



SpiNNaker-1 machine

Many-core system
0.5 (1.0) Million ARM cores
Real-time simulator

BrainScaleS-1 machine

Physical model system
4M neurons, 1B plastic syn.
Accelerated emulator

SpiNNaker-2 prototype

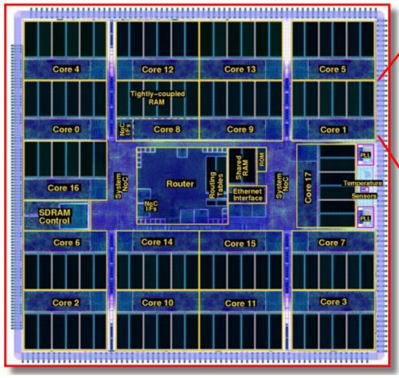
144 Cortex M4F per chip
36 GIPS/Watt per chip
x10 with constant power

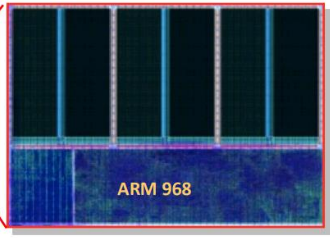
BrainScaleS-2 prototype

On-chip plasticity processor
Flexible hybrid plasticity
Active dendritic spatial structure

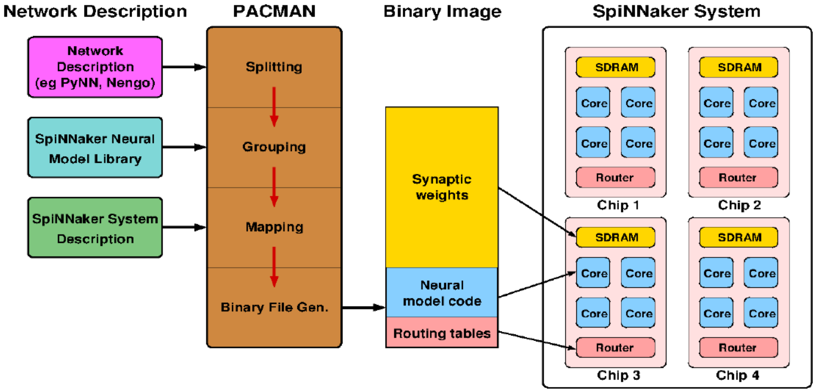
https://www.humanbrainproject.eu/en/

Healthcare AI-Chips





SpiNNaker CPU



Network Description: Network Description (eg PyNN, Nengo); SpiNNaker Neural Model Library; SpiNNaker System Description

PACMAN: Splitting → Grouping → Mapping → Binary File Gen.

Binary Image: Synaptic weights; Neural model code; Routing tables

SpiNNaker System: Chip 1, Chip 2, Chip 3, Chip 4 — each with SDRAM, Core, Core, Core, Core, Router

SpiNNaker-1 machine

Many-core system
0.5 (1.0) Million ARM cores
Real-time simulator

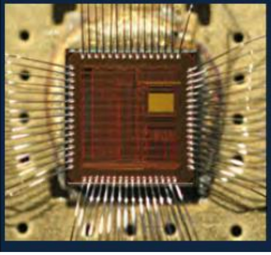
BrainScaleS-1 machine

Physical model system
4M neurons, 1B plastic syn.
Accelerated emulator

SpiNNaker-2 prototype

144 Cortex M4F per chip
36 GIPS/Watt per chip
x10 with constant power

BrainScaleS-2 prototype



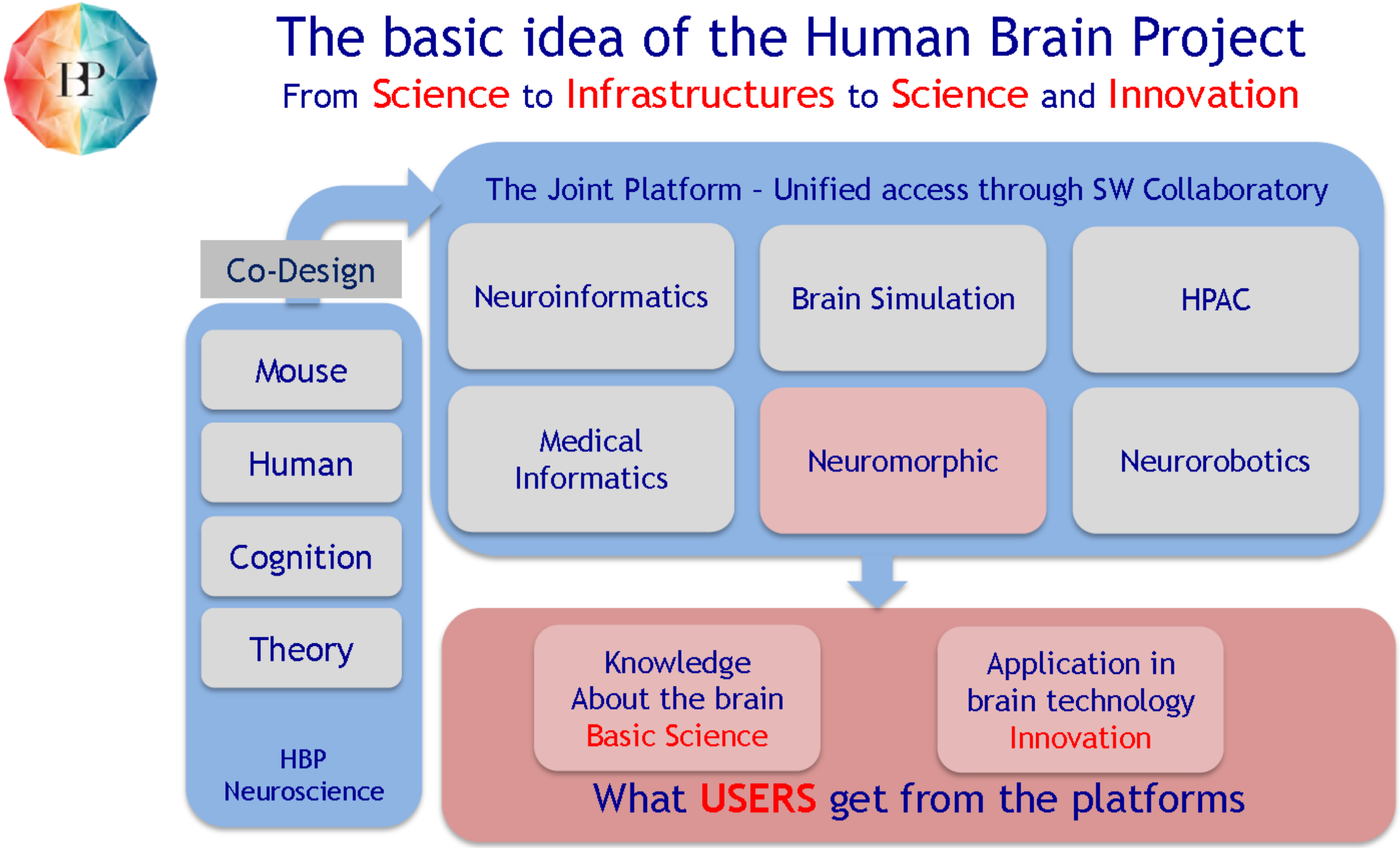
On-chip plasticity processor
Flexible hybrid plasticity
Active dendritic spatial structure

https://www.humanbrainproject.eu/en/

Healthcare AI-Chips

The Human Brain Project

An EU ICT Flagship project (€1B budget)
80 partner institutes, led by Henry Markram, EPFL


HBP
The basic idea of the Human Brain Project
From Science to Infrastructures to Science and Innovation
Co-Design
Mouse
Human
Cognition
Theory
HBP Neuroscience
The Joint Platform - Unified access through SW Collaboratory
Neuroinformatics
Brain Simulation
HPAC
Medical Informatics
Neuromorphic
Neurorobotics
Knowledge About the brain Basic Science
Application in brain technology Innovation
What USERS get from the platforms

https://www.humanbrainproject.eu/en/

..our work - Homeostatic Neuromorphic System

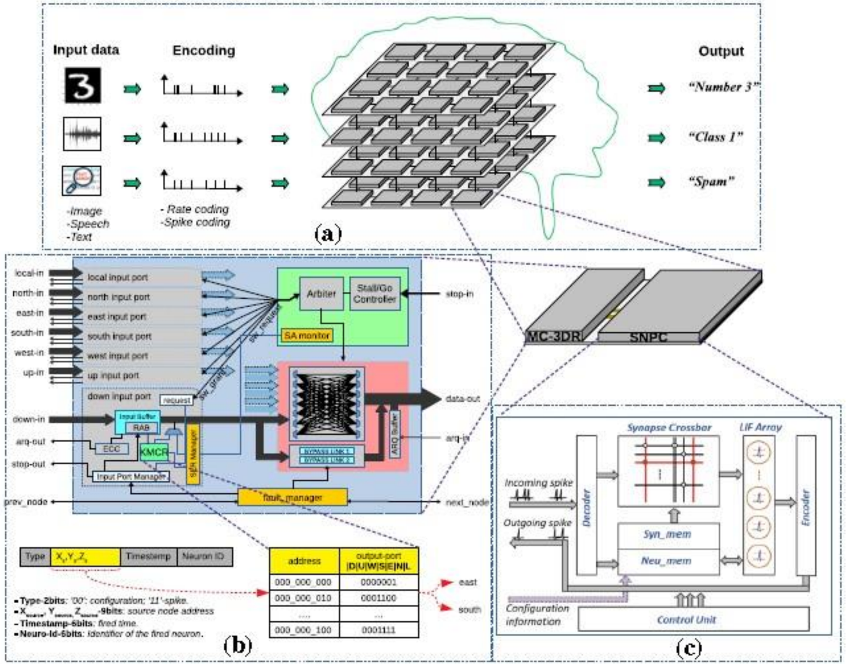
*this is not the scope of this talk

Our work - Homeostatic Neuromorphic System

Architecture: Spike Packet Format

2 bits	3 bits	9 bits	6 bits	8 bits
Type	**[Fault_flag]**	**XYZ_s**	**Timestamp**	**Neuron ID**

- Type: It is the header of the packet indicating this packet is either for configuration or spike: ’00’: system configuration; ’11’, spike packet.
- [Fault_Flag]: This is only used for the fault-tolerant multicast routing algorithm
- XYZ_s: It is the address of the source neuron tile, used for spike routing.
- Timestamp: In spiking neuron network, the time of the generated spike is used to encode the information.
- Neuron ID: this is the identifier of the pre-synaptic neuron.



3DNoC-SNN system architecture high-level view.

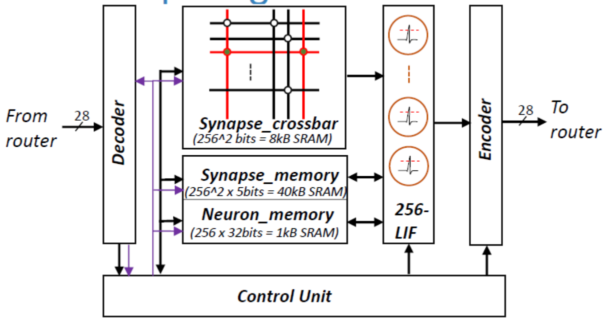
Table 5: Power consumption of the KMCR and FTSP-KMCR under the benchmarks.

System	KMCR		FTSP-KMCR	
	Inv. Pen.	Wis.	Inv. Pen.	Wis.
Area (mm^2)	0.102	0.346	0.108	0.365
Power (mW)	10.13	34.20	10.64	35.92

Table 6: MC-3DR Hardware Complexity Evaluation and Comparison.

System	Topology	Area (mm^2)	Power (mW)
EMBRACE router [Carrillo2012], 90nm	2D Mesh	0.056	1.72
HANA tile router [Liu2016], 90nm	2D Mesh	0.156	28.12
H-NoC cluster router [Crrillo2012HNoC], 65nm	Star-Mesh	0.022	1.19
Clos-NoC spine switch [Hojabr2017], 45nm	Custom Clos	0.076	-
Clos-NoC leaf switch [Hojabr2017], 45nm	Custom Clos	0.061	-
MC-3DR router, 45nm (this work)	3D Mesh	0.031	1.66

Architecture: Spiking Neural Processing Core



5bits synapse register format

Input type [0]	Synaptic strength [1:4]

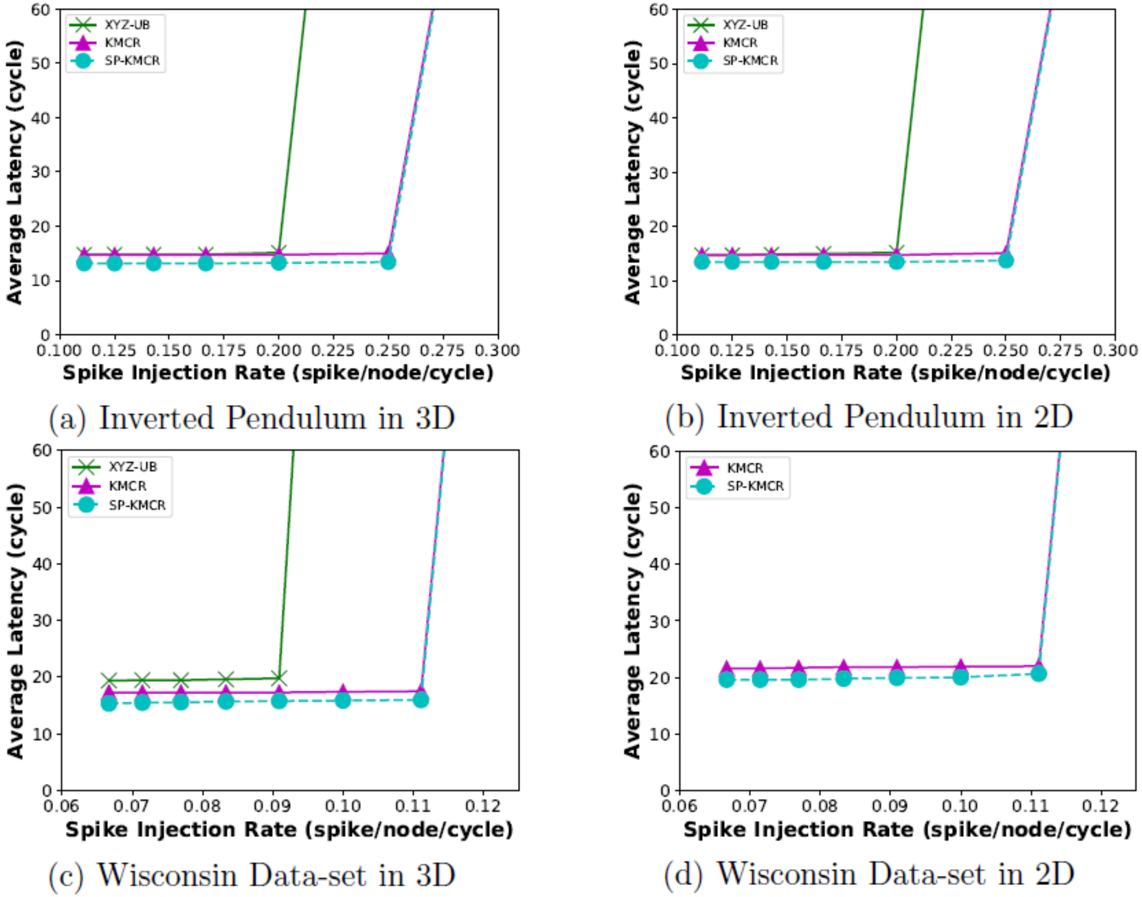
32bits neuron register format

Membrane potential [0:7]	Threshold [8:15]	Leaky value [16:23]	Reset value [24:31]

Figure 12: Spiking Neuron Processing Core (SNPC) architecture [1]

Our work - Homeostatic Neuromorphic System

Average spike latency over varying the injection rate



(a) Inverted Pendulum in 3D

(b) Inverted Pendulum in 2D

(c) Wisconsin Data-set in 3D

(d) Wisconsin Data-set in 2D

Figure 15: Average packet latency evaluation result

○The H. Vu,Yuichi Okuyama, Abderazek Ben Abdallah, "Comprehensive Analytic Performance Assessment and K-means based Multicast Routing Algorithms and Architecture for 3D-NoC of Spiking Neurons," *ACM Journal on Emerging Technologies in Computing Systems (JETC), Special Issue on Hardware and Algorithms for Learning On-a-chip for Energy-Constrained On-Chip Machine Learning, Vol. 15, No. 4, Article 34, October 2019. doi: 10.1145/3340963*

○The H. Vu, Ogbodo Mark Ikechukwu, and Abderazek Ben Abdallah, "Fault-tolerant Spike Routing Algorithm and Architecture for Three Dimensional NoC-Based Neuromorphic Systems'", *IEEE Access, vol. 7, pp. 90436-90452, 2019.*

Agenda

- **Fundamental Trends**
- **AI – The 4th Industrial Revolution**
- **Survey of AI Hardware**
  - ➢ Cloud AI Hardware, Chips
  - ➢ Mobile AI Chips
  - ➢ Edge and IoT AI Chips
  - ➢ Healthcare AI Chips
- **Conclusions**

Conclusions

- DNNs are a key component in the AI revolution.
- Efficient processing of DNNs is an important area of research with many promising opportunities for innovation at various levels of hardware design, including algorithm co-design
- It's important to consider a comprehensive set of metrics when evaluating different DNN solutions: **accuracy, speed, energy, and cost**

Conclusions

Memory access in AI-Chip is the bottleneck

- **Worst case:** ALL memory R/W are DRAM accesses

Ex. AlexNet [NIPS 2012] has 724M MACs → **2896M DRAM accesses required**

Possible HW/SW techniques to cope with the memory access problem:

❖**Advanced Storage Technology**

- Embedded DRAM (eDRAM) → Increase on-chip storage capacity
- 3D Stacked DRAM → Increase memory bandwidth
- Use memristors as programmable weights (resistance)

❖**Reduce size of operands for storage/compute**

- Floating point → Fixed point
- Bit-width reduction

❖**Reduce number of operations for storage/compute**

- Network Pruning; Compact Network Architectures

References

1. Dally, W. February 9, 2016. High Performance Hardware for Machine Learning, Cadence ENN Summit
2. Michael Alba, The Great Debate of AI Architecture, April 2018 [www.engineering.com].
3. [Ros15a] Ross, J., Jouppi, N., Phelps, A., Young, C., Norrie, T., Thorson, G., Luu, D., 2015. Neural Network Processor, Patent Application No. 62/164,931.
4. Jiantao Qiu, Jie Wang, Song Yao, Kaiyuan Guo, Boxun Li, Erjin Zhou, Jincheng Yu, Tianqi Tang, Ningyi Xu, Sen Song, Yu Wang, Huazhong Yang, Going Deeper with Embedded FPGA Platform for Convolutional Neural Network, ACM International Symposium on FPGA, 2016
5. Dally, NIPS'2016 workshop on Efficient Methods for Deep Neural Networks
6. https://itcafe.hu/dl/cnt/2017-12/142233/idc_white_paper.pdf
7. Top AT Trends to watch in 2018, CBINSIGHTS, 2018
8. What is TensorFlow? | Introduction to TensorFlow | TensorFlow Tutorial for Beginners | Simplilearn https://www.youtube.com/watch?v=E8n_k6HNAgs
9. TensorFlow in 5 Minutes (tutorial) https://www.youtube.com/watch?v=2FmcHiLCwTU
10. Hardware Architectures for Deep Neural Networks, ISCA Tutorial June 24, 2017, http://eyeriss.mit.edu/tutorial.html
11. Quantifying the performance of the TPU, our first machine learning chip: https://cloudplatform.googleblog.com/2017/04/quantifying-the-performance-of-the-TPU-our-first-machine-learning-chip.html
12. https://streamable.com/
13. Abderazek Ben Abdallah, "Neuro-inspired Computing Systems & Applications", Keynote Speech, 2018 International Conference on Intelligent Autonomous Systems (ICoIAS'2018), March 1-3, 2018, Singapore.[slides.pdf]
14. The H. Vu, Ryunosuke Murakami, Yuichi Okuyama, Abderazek Ben Abdallah, "Efficient Optimization and Hardware Acceleration of CNNs towards the Design of a Scalable Neuro-inspired Architecture in Hardware", Proc. of the IEEE International Conference on Big Data and Smart Computing (BigComp-2018), pp. 326-332, January 15-18, 2018, Shanghai, China. [paper.pdf], [slides.pdf]
15. Ryunosuke Murakami, Yuichi Okuyama, Abderazek Ben Abdallah, "Animal Recognition and Identification with Deep Convolutional Neural Networks for Farm Monitoring", Information Processing Society Tohoku Branch Conference, Feb. 10, 2018 [slides.pdf]
16. Yuji Murakami, Yuichi Okuyama, Abderazek Ben Abdallah, "SRAM Based Neural Network System for Traffic-Light Recognition in Autonomous Vehicles", Information Processing Society Tohoku Branch Conference, Feb. 10, 2018. [slides.pdf]
17. Kanta Suzuki, Yuichi Okuyama, Abderazek Ben Abdallah, "Hardware Design of a Leaky Integrate and Fire Neuron Core Towards the Design of a Low-power Neuro-inspired Spike-based Multicore SoC", Information Processing Society Tohoku Branch Conference, Feb. 10, 2018. [slides.pdf]
18. Spiking Neuron Models Single Neurons, Populations, Plasticity Wulfram Gerstner and Werner M. Kistler Cambridge University Press, 2002