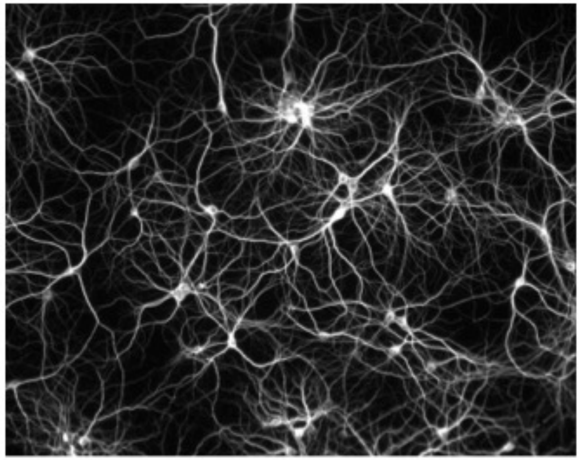
Multilayer spintronic neural networks with radio-frequency connections

Alice Mizrahi[1]

Nathan Leroux[1], Andrew Ross[1], Danijela Marković[1], Juan Trastoy[1], Damien Querlioz[2], Leandro Martins[3], Alex Jenkins[3], Ricardo Ferreira[3], Julie Grollier[1]

[1]CNRS/THALES, FRANCE [2]C2N, FRANCE [3]INL,PORTUGAL

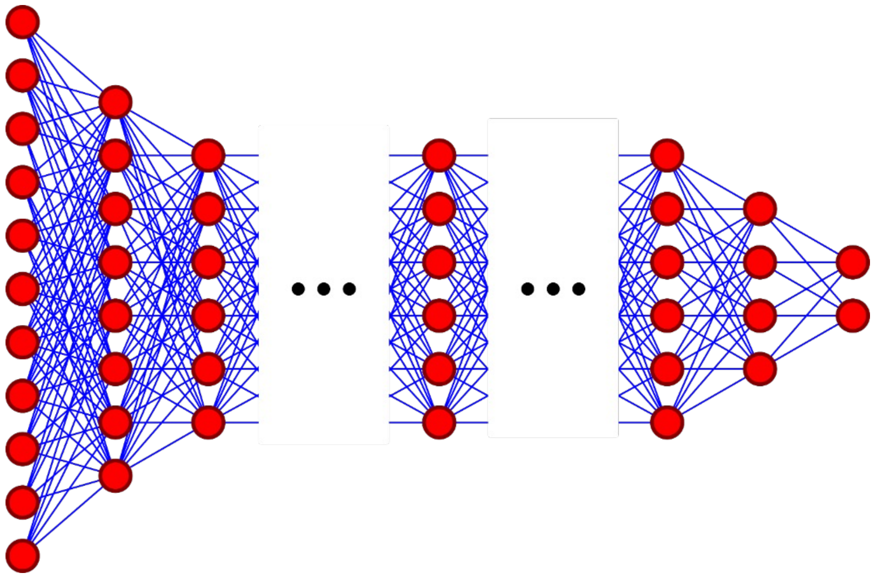
The brain is a source of inspiration for energy efficiency



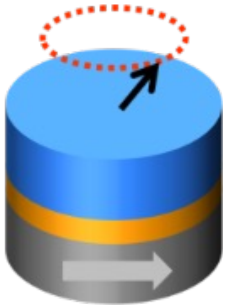
GPT-3 : understanding and generating text, images ← ***Energy budget 190 000 kWh*** → Brain : 1000 years of operation

Marković, D., Mizrahi, A., Querlioz, D., & Grollier, J. (2020). Physics for neuromorphic computing. *Nature Reviews Physics*, 2(9), 499-510.

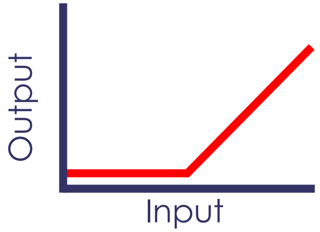
Challenge one: reproduce essential features of synapses and neurons with nanoscale devices



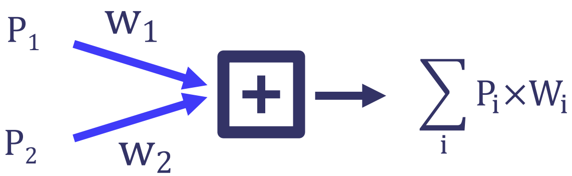
Radiofrequency Magnetic Tunnel Junctions as **both** neurons and synapses



Neurons
Non-linear activation function

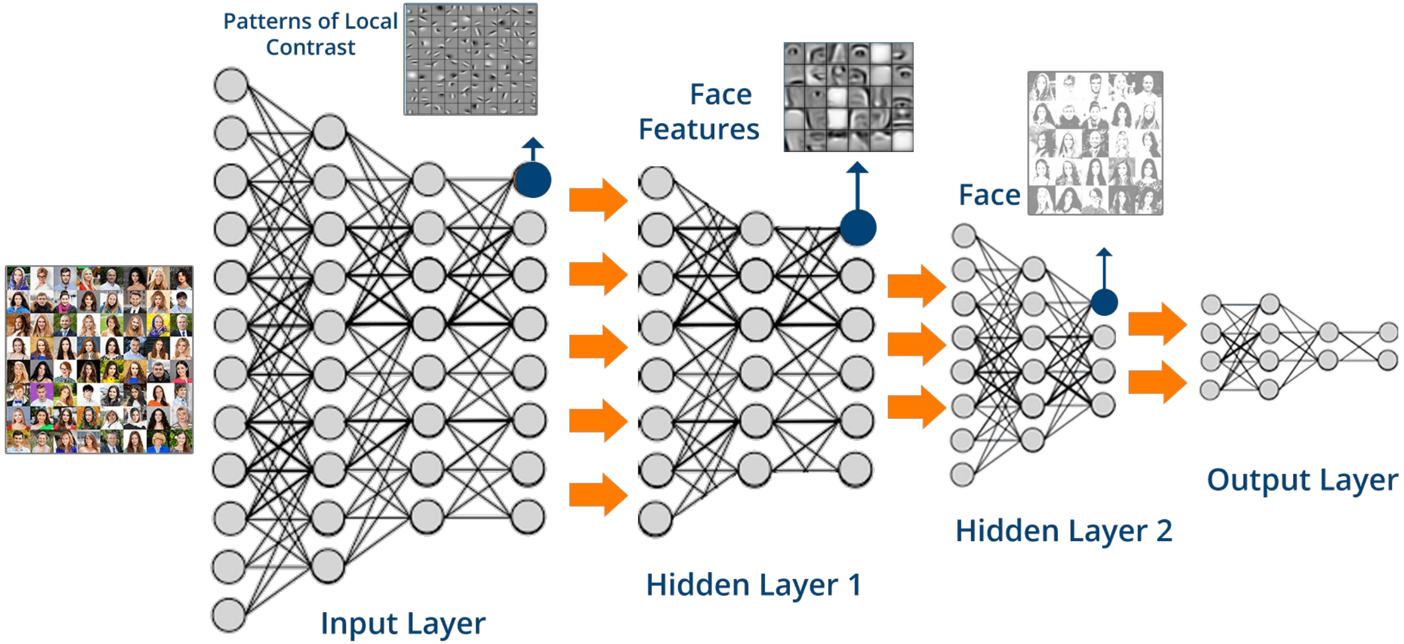


Synapses: memory
Weighted sum

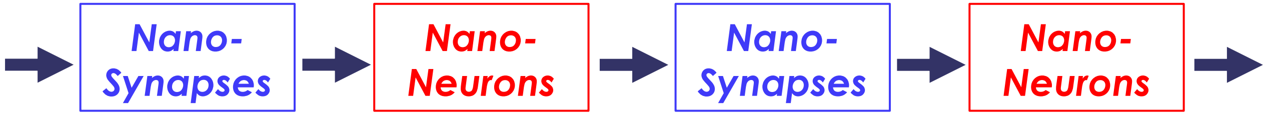


THALES

Challenge two: build deep modular networks



Depth is a requirement for complex tasks (100+ layers)
=> We need to be able to stack layers

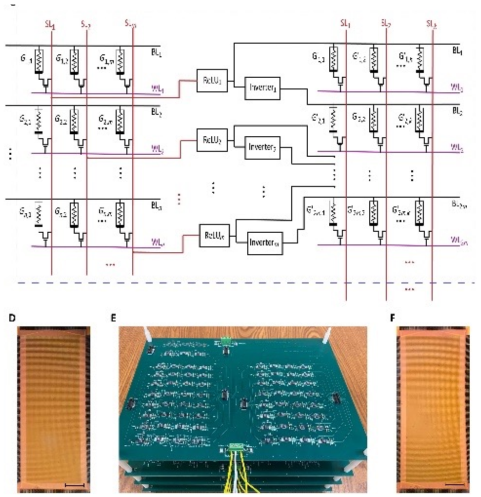


How to feed the output of a device to the next one?
How to co-integrate the different devices?

But the outputs of nanoneurons are often no adapted

Nano-synapses connected by off-the-shelf neuron circuits

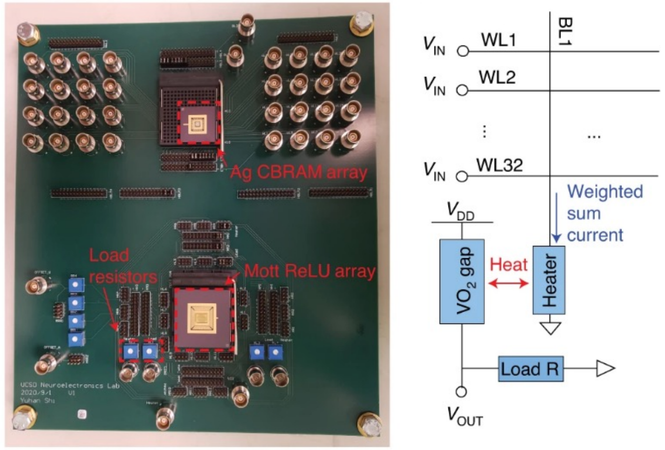
Nano-Synapses → *Not nano* → *Nano-Synapses*



Kiani et *al.*, *Sci. Adv.*, 7, 48 (2021)

Nano-synapses connected **to** nano-neurons

Nano-Synapses → *Nano-Neurons*

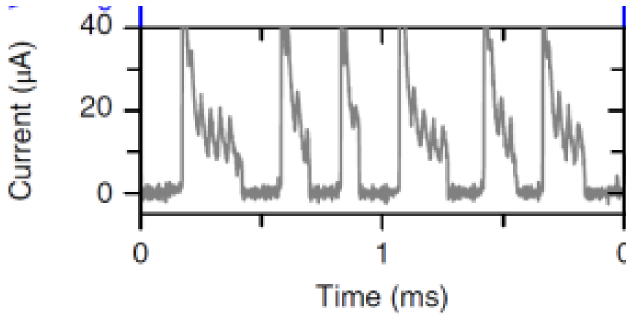


Oh *et al.*, *Nat. Nano.* 16 (2021)

Connecting nano-neurons to nano-synapses is a challenge

Nano-Neurons → *Nano-Synapses*

Because nano-neuron outputs tend to need reshaping

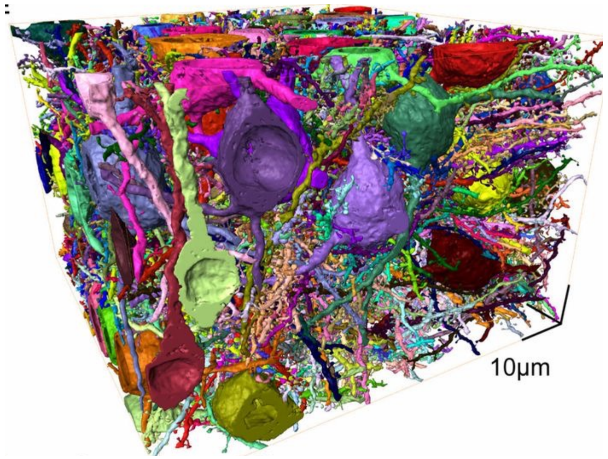


Z. Wang et al, Nature Electronics 1, 137 (2018)

THALES

Challenge three: build densely connected networks

Brain: 10^4 synapses/neurons
AI: thousands synapses/neurons



Motta et al, Science, 366, 6469 (2019)

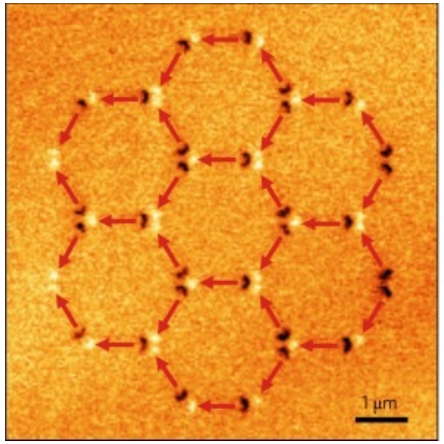
We need **long range** connections
We need **high density** of connections

Connections must be **independently tunable**

THALES

But most physical systems are naturally locally connected

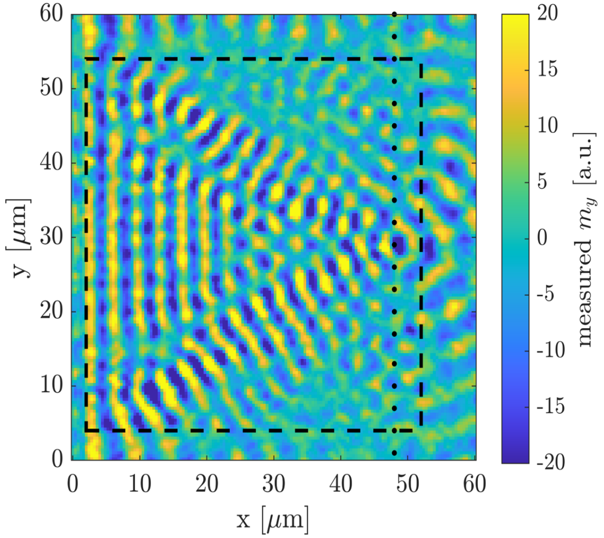
Spin ice lattice



J. C. Gartside et al, Nature Nano 13, 53 (2018)

Each magnet is connected to 4 others

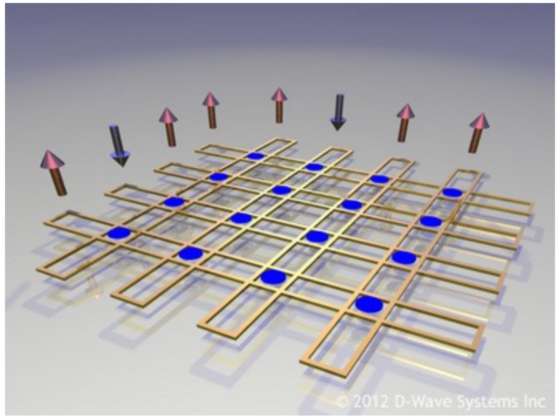
Spin waves

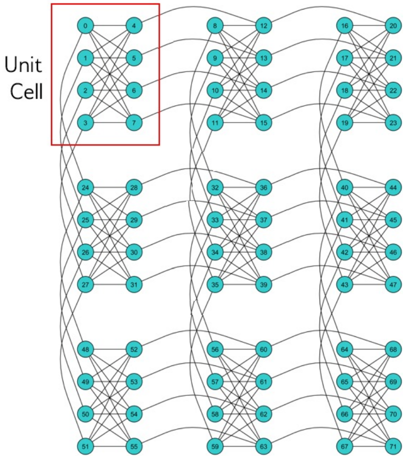


M Kiechle et al, arXiv:2207.00055

Local tunability of spin-wave ≠ all-to all tunable connectivity

D Wave Ising machine with squid Qubits

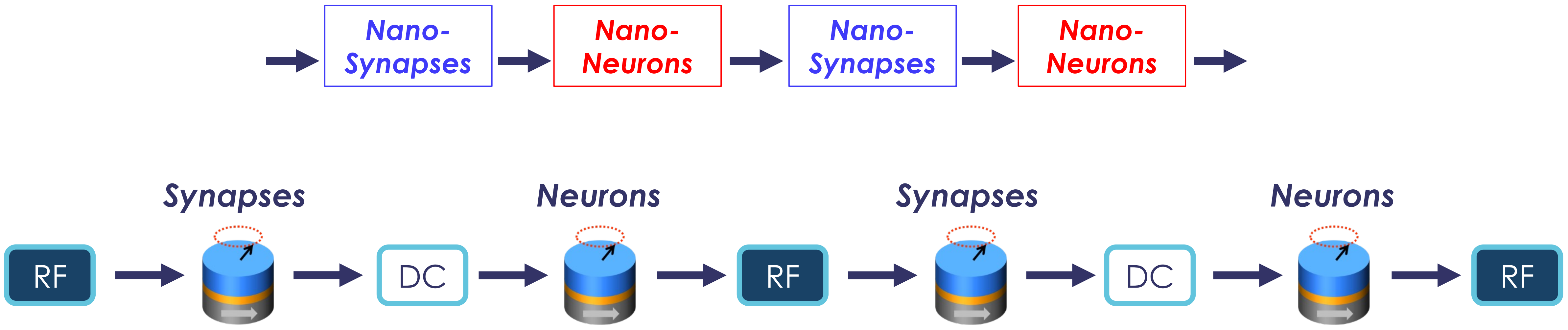




8 qubits connected all to all through JJs

2000 spins in total, but only ~50 spins can be coupled all to all through embedding

We build a fully spintronic RF multilayer neural network



⇒Same device for neurons and synapses

⇒Stack layers alternating DC and RF for depth

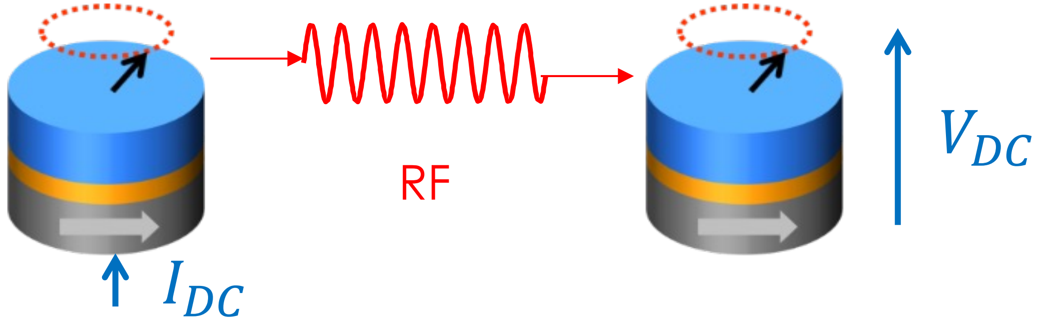
⇒Dense connectivity with RF signals using frequency multiplexing

We experimentally demonstrate the connections between layers and build a network

THALES

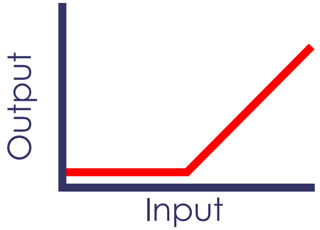
RF magnetic tunnel junctions as both neurons and synapses

Neuron:
DC to RF
conversion

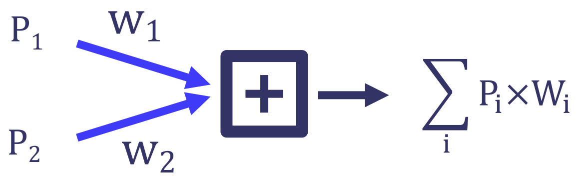


Synapse:
RF to DC
conversion

Non-linear activation function

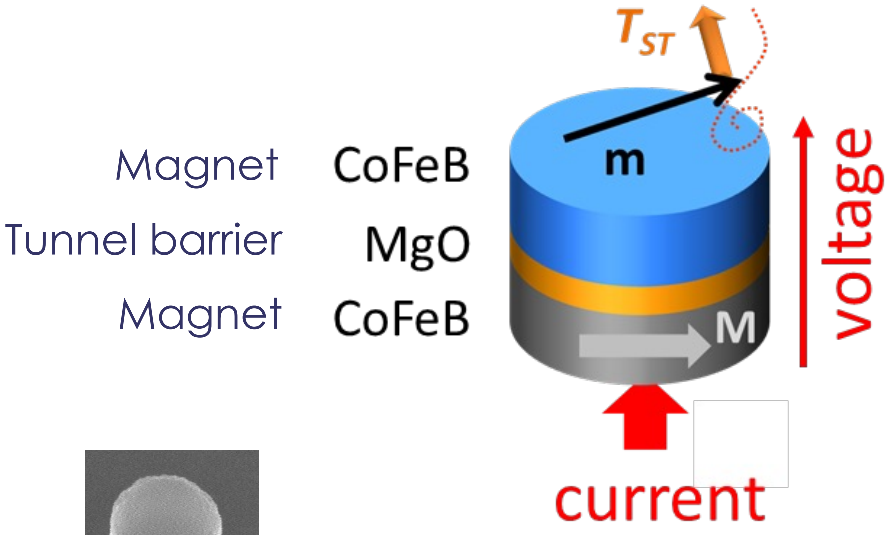


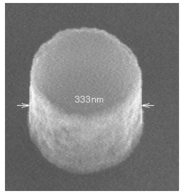
Weighted sum



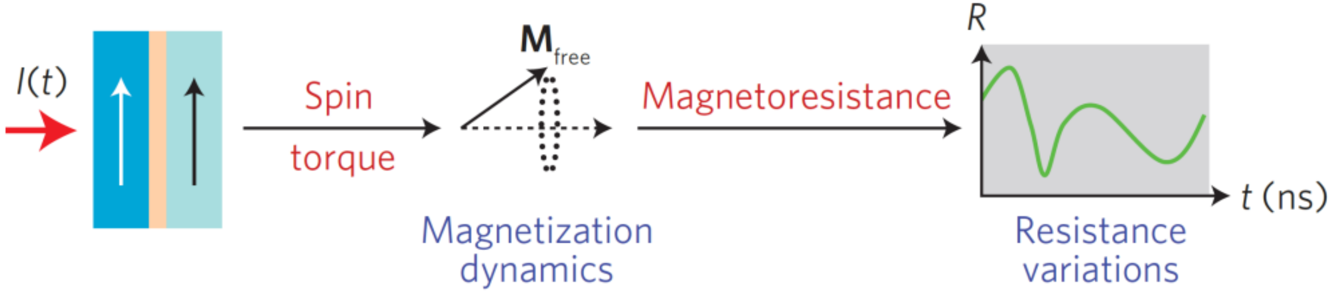
Could be any pair of oscillator - resonator

The Magnetic Tunnel Junction: multifunctional mature technology





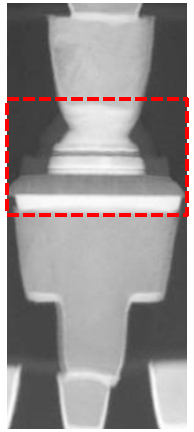
Electrical Read and Write



J. Grollier et al, PIEEE 104, 2024 (2016)

Small size
Non-volatility
Endurance
Reproducibility
CMOS compatibility
Maturity

Samsung IEDM 2018, 28nm Logic

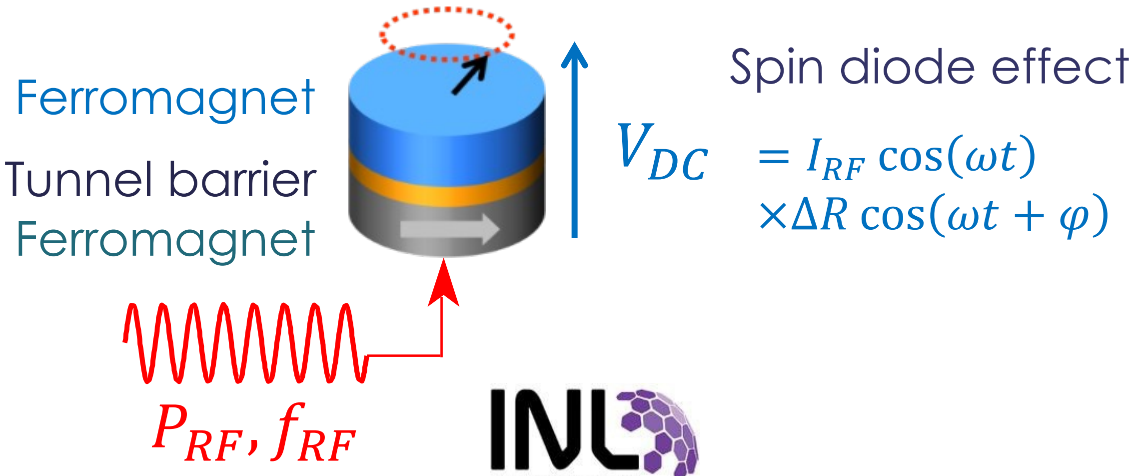


Gbit embedded memory chip commercially available (MRAM)

Magnetic tunnel junctions as synapses

THALES

Magnetic tunnel junctions multiply input RF signals by a synaptic weight

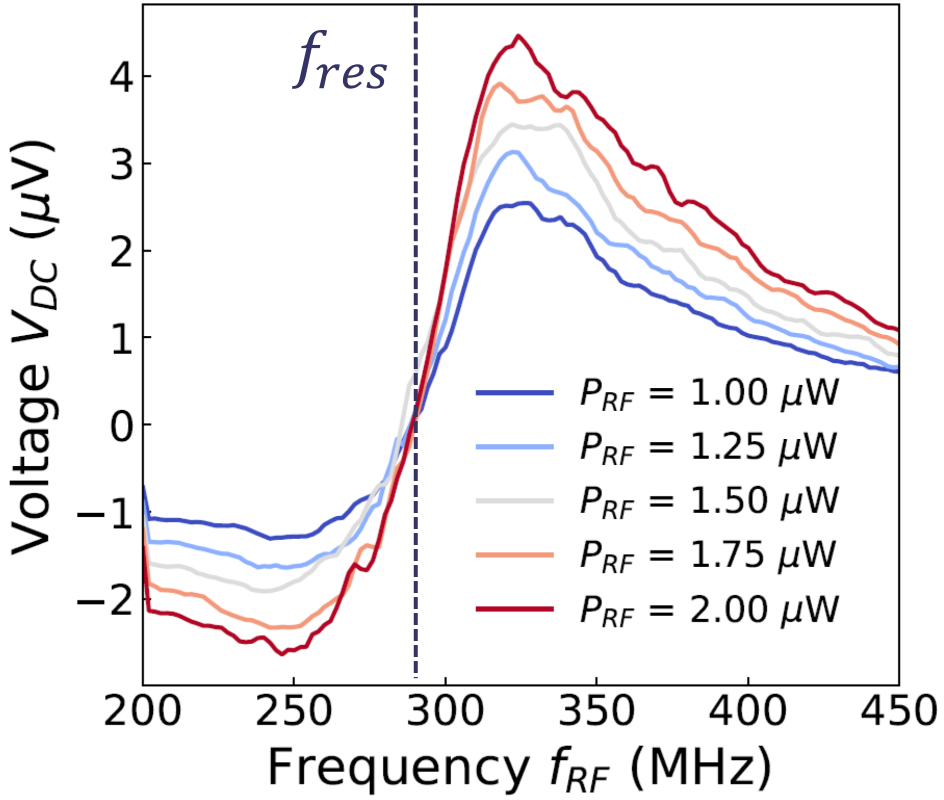


INTERNATIONAL IBERIAN NANOTECHNOLOGY LABORATORY

$$V_{DC} = P_{RF} \times W(f_{RF} - f_{res})$$

Output = Input × Weight

Experiments

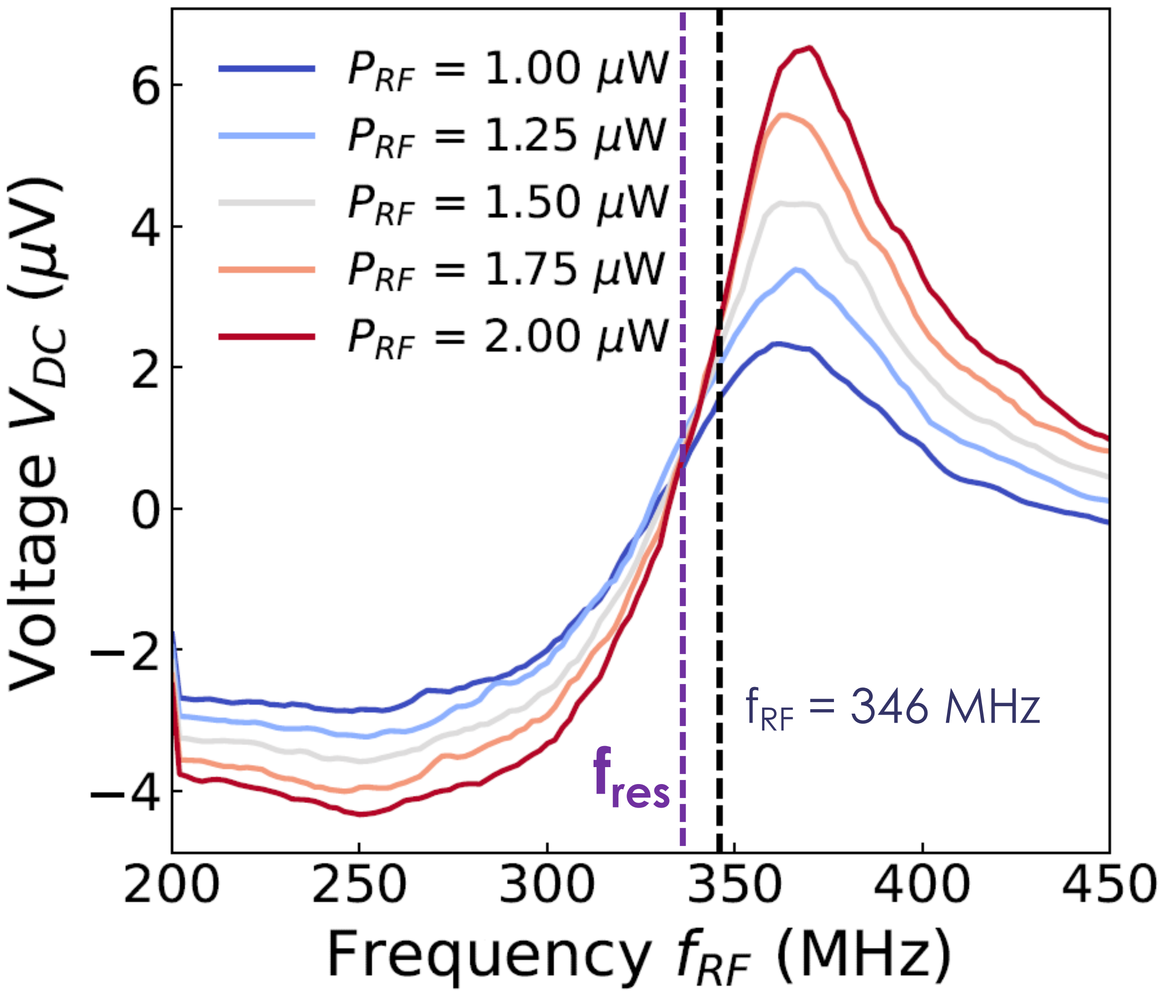


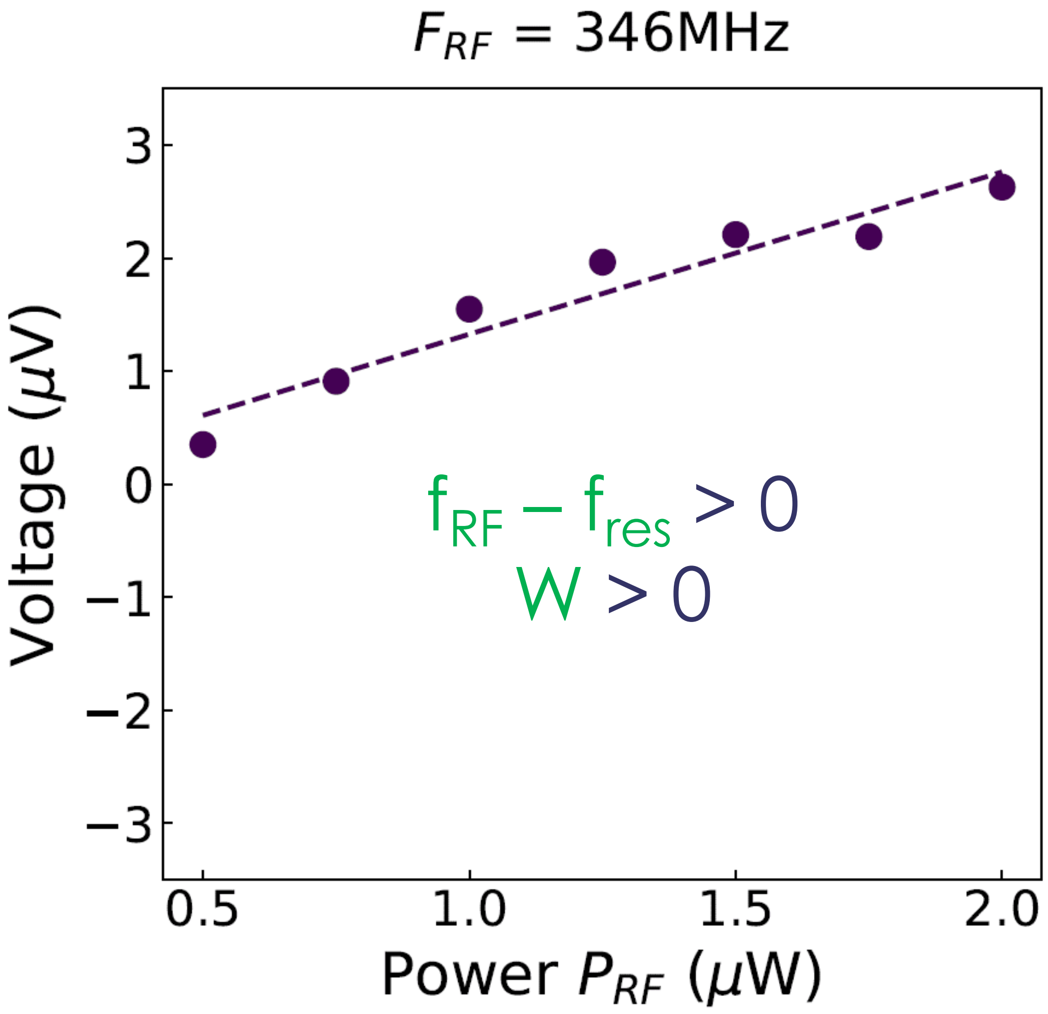
Nathan Leroux, Alice Mizrahi et al, *Physical Review Applied* 15, 034067 (2021)

THALES GROUP OPEN

THALES

The MTJ multiplies the RF power by a synaptic weight

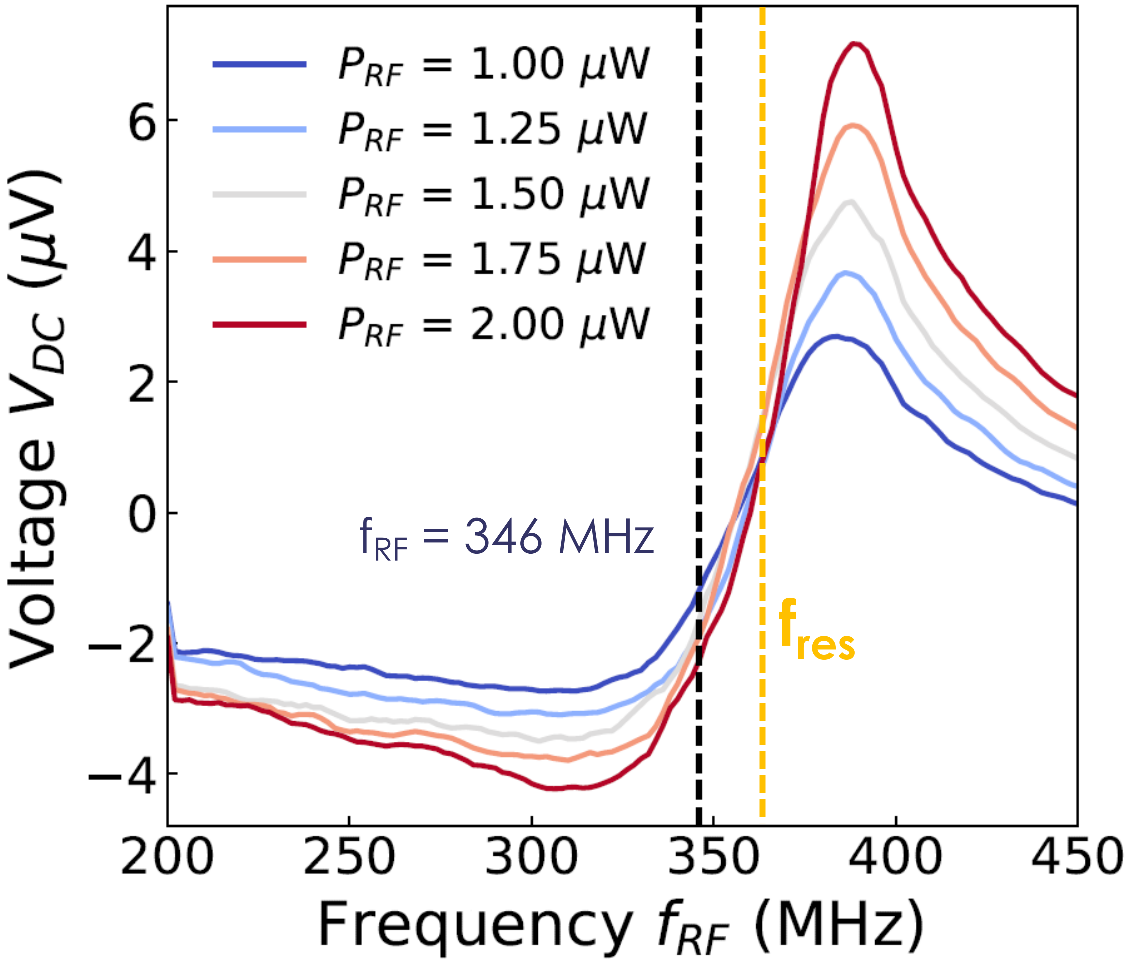


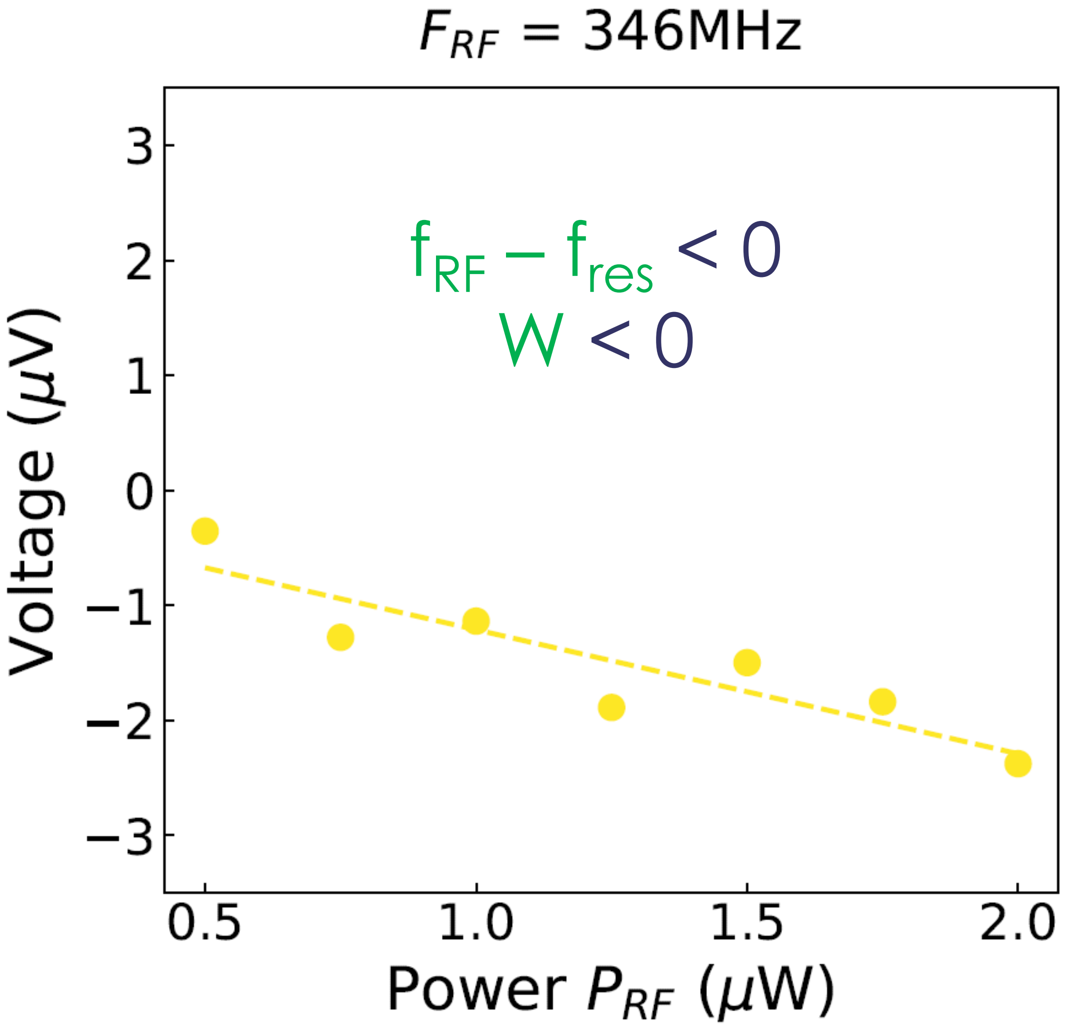


Synaptic multiplication : Output = W x Input

$$V = W(f_{RF} - f_{res}) \times P_{RF}$$

The MTJ multiplies the RF power by a synaptic weight



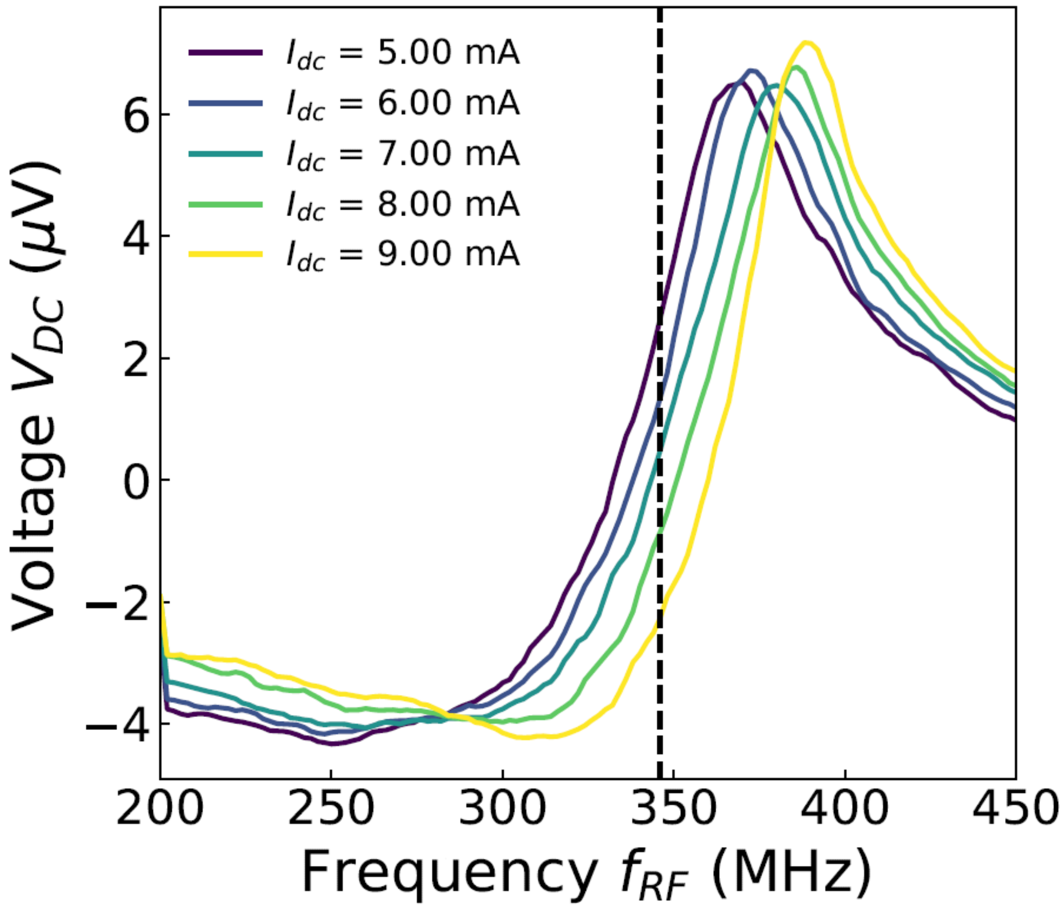


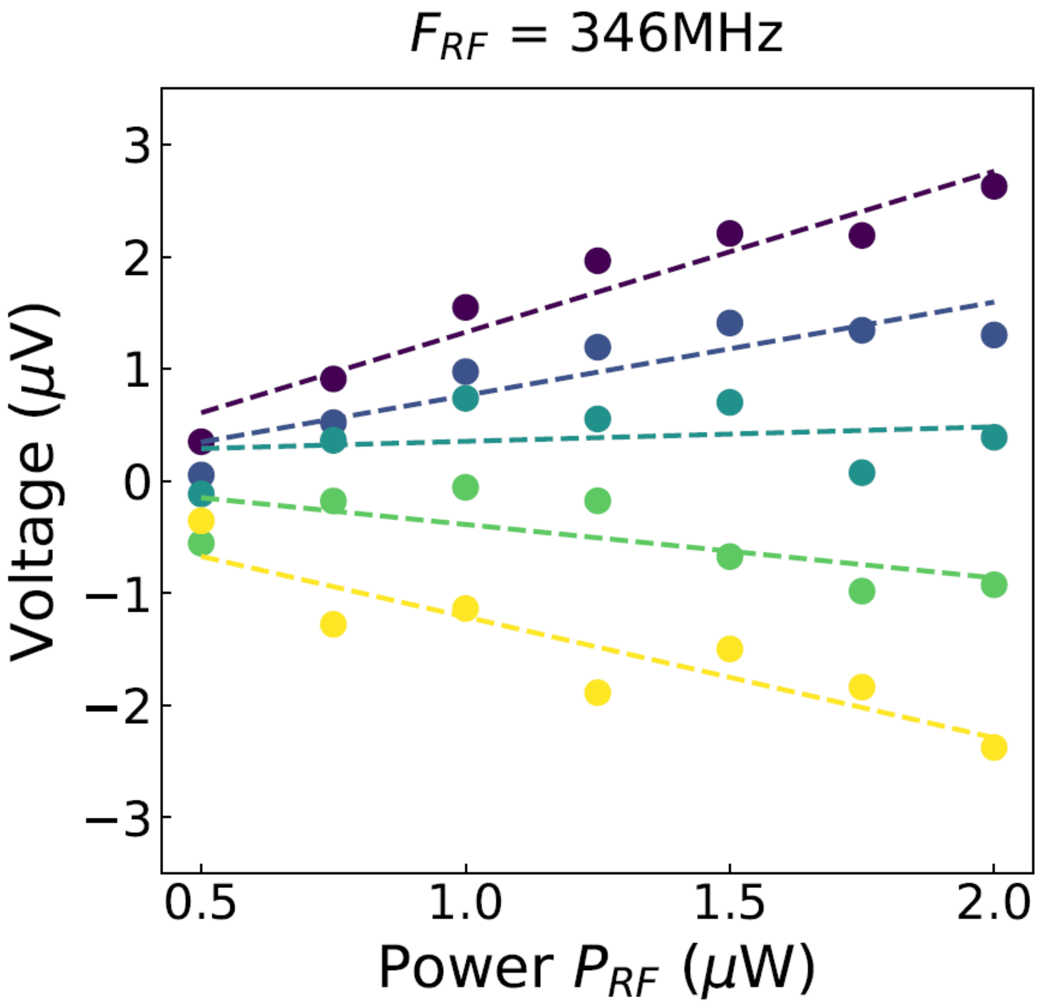
Synaptic multiplication : Output = W x Input

$$V = W(f_{RF} - f_{res}) \times P_{RF}$$

THALES

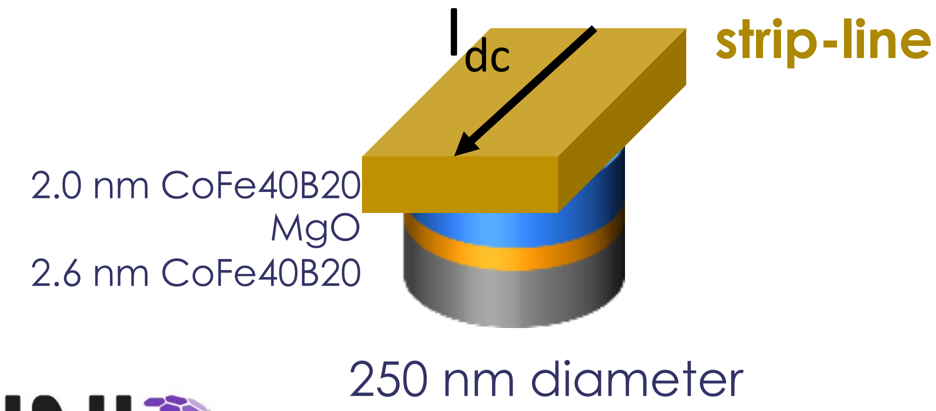
We can continuously tune the synaptic weight by tuning the resonance frequency



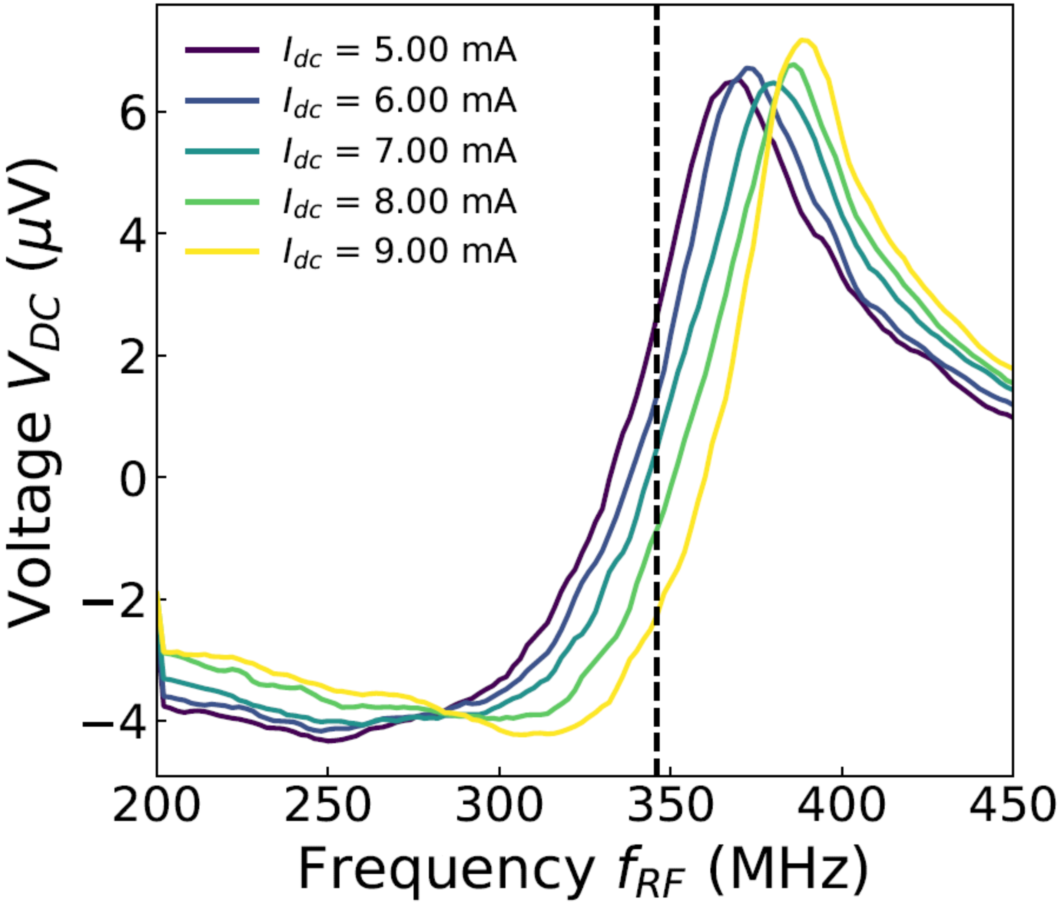


Tuning the resonance frequency

Here: control by local magnetic field



INL INTERNATIONAL IBERIAN NANOTECHNOLOGY LABORATORY

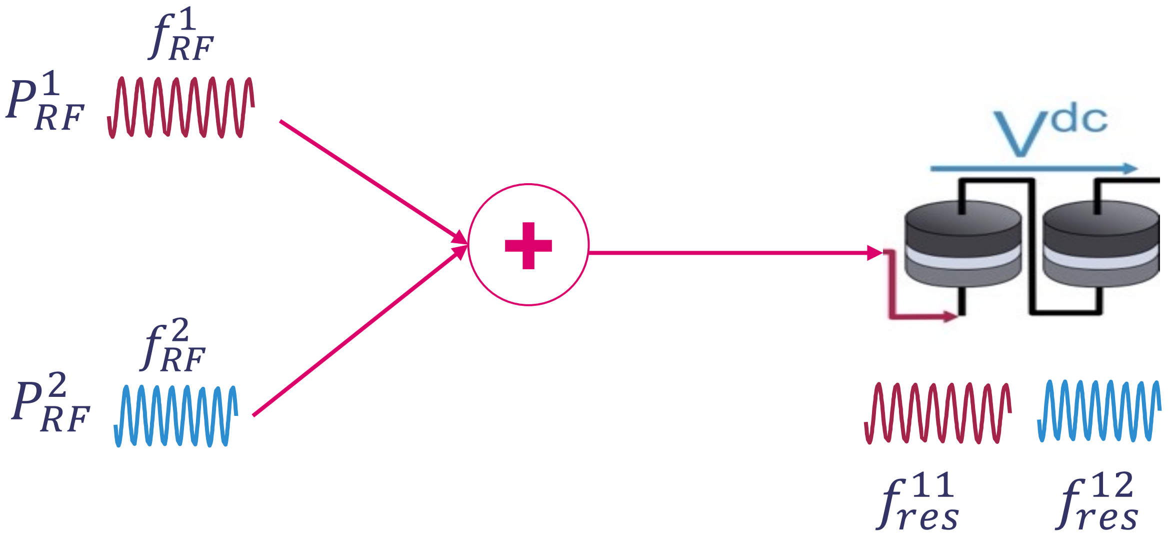


In the future: **non-volatile control** of frequency through magnetic anisotropy by resistive switch material

M. Zahedinejad et al., Nature Materials (2021).

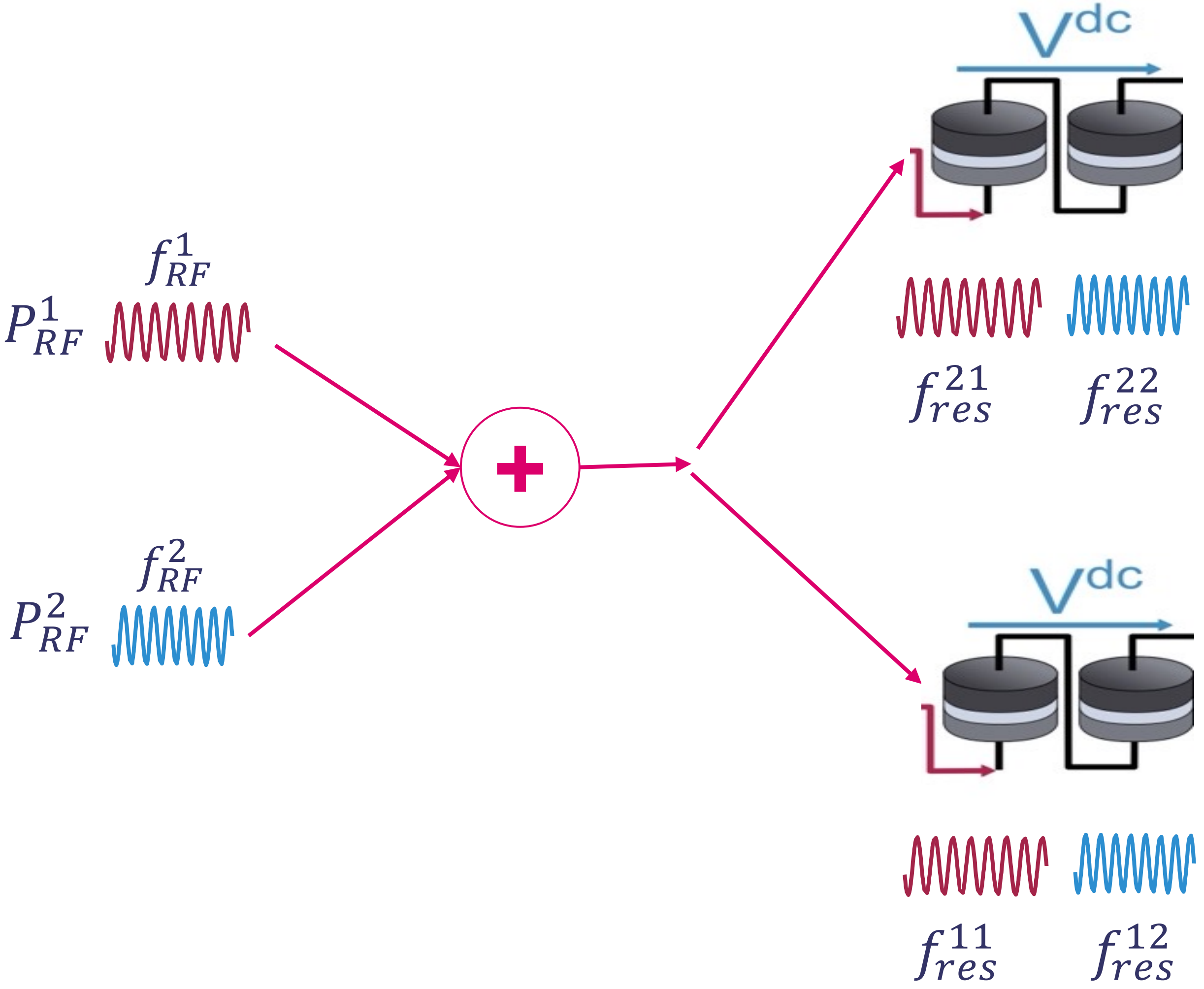
THALES

Weighted sum through frequency multiplexing



$$V_1 = \sum_i P_{RF}^i W(f_{res}^{1i})$$

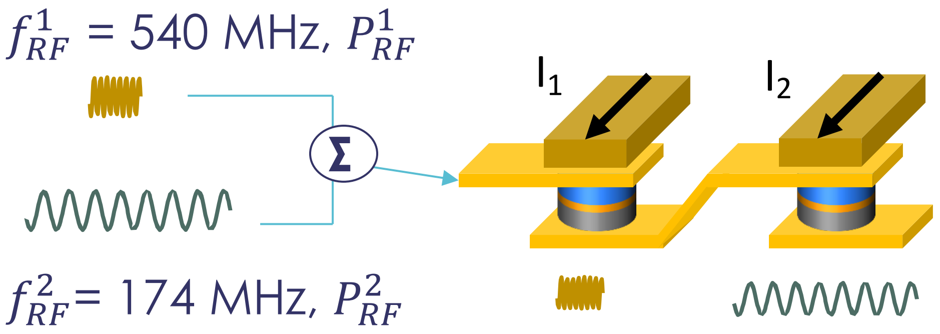
Frequency multiplexing make high density connectivity possible



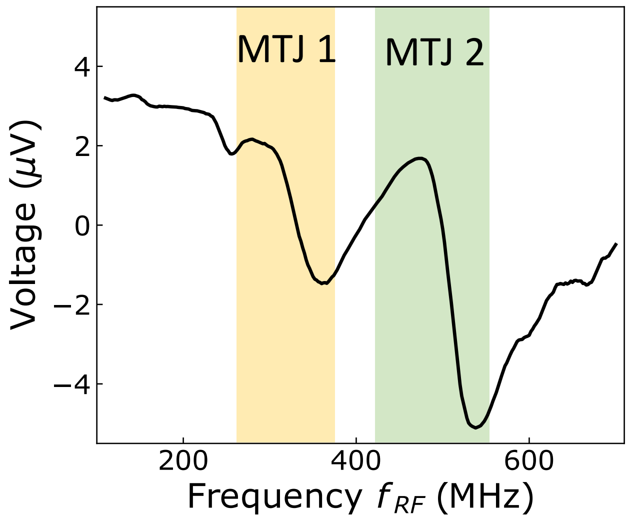
$$V_2 = \sum_i P_{RF}^{i} W(f_{res}^{2i})$$

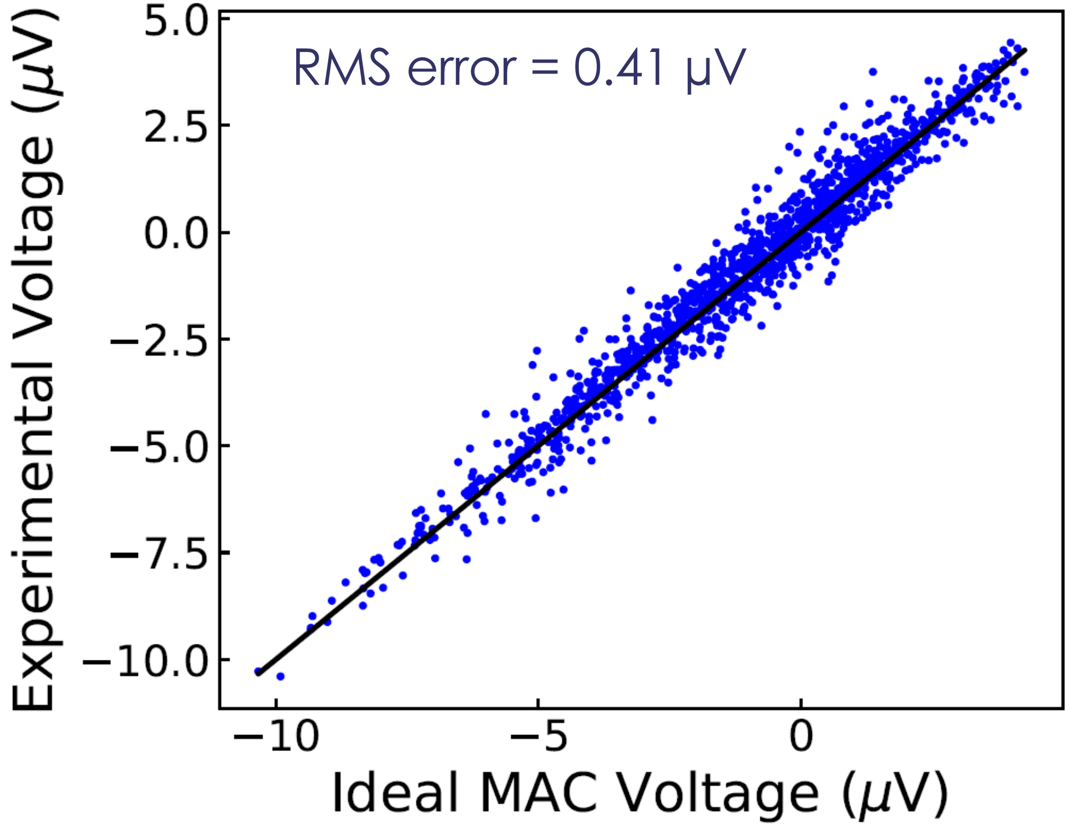
$$V_1 = \sum_i P_{RF}^{i} W(f_{res}^{1i})$$

Two magnetic tunnel junctions perform the weighted sum on raw analogue RF signals



$$V_{th} = P_{RF}^1 \times W_1(f_{RF}^1 - f_{res}^1) + P_{RF}^2 \times W_2(f_{RF}^2 - f_{res}^2)$$

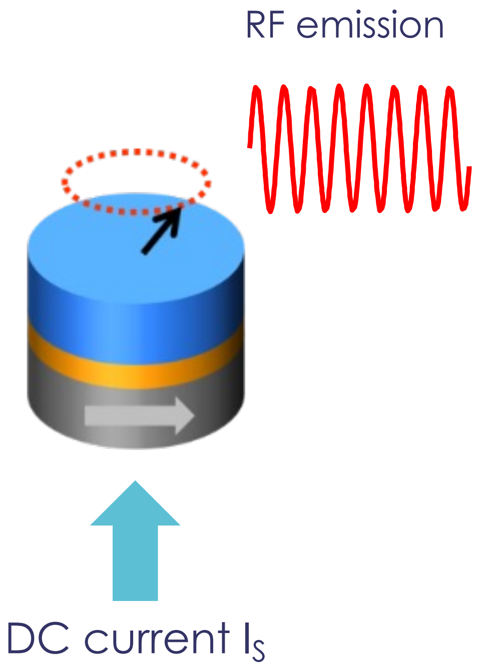


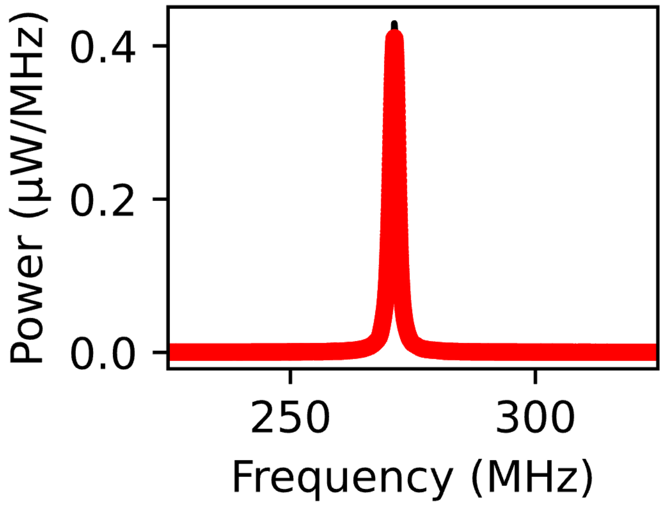


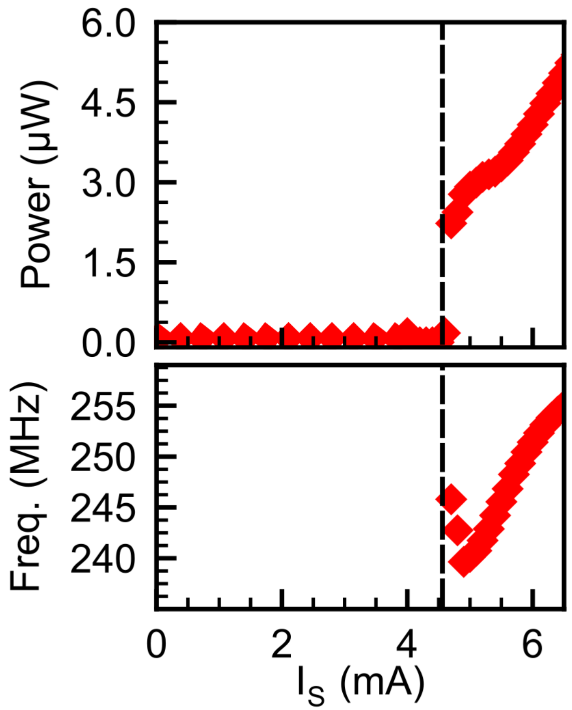
Leroux et al, "Hardware realization of the multiply and accumulate operation on radio-frequency signals with magnetic tunnel junctions." *Neuromorphic Computing and Engineering* (2021).

Magnetic tunnel junctions as neurons

MTJs as radiofrequency neurons





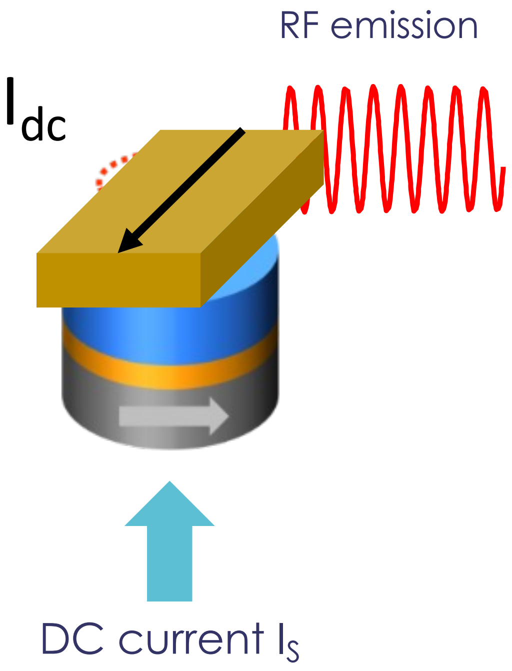


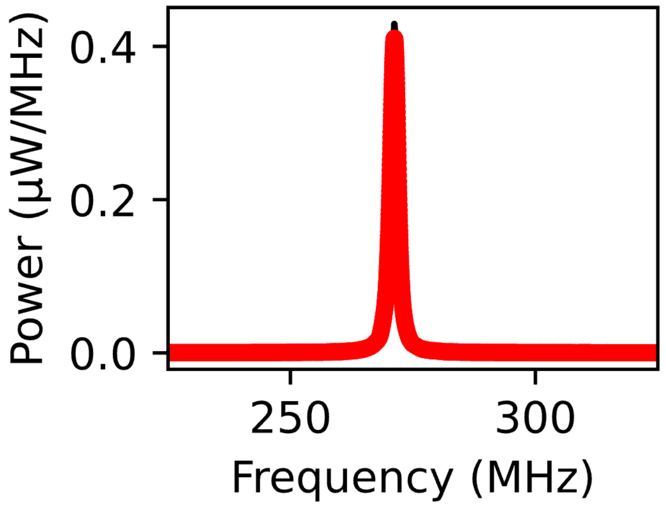
- Radio Frequency Emissions above threshold current
- Emitted power follows activation function:

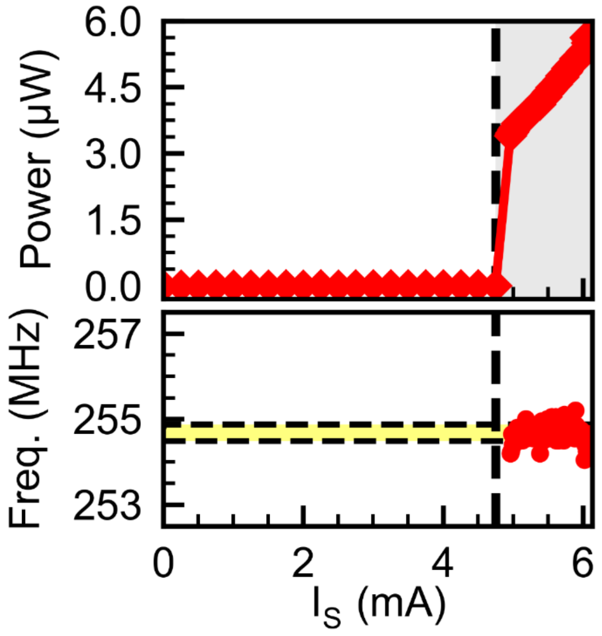
$$\phi(\boldsymbol{v}) = \max(0, \boldsymbol{a} + \boldsymbol{v}'\boldsymbol{b})$$

THALES

MTJs as radiofrequency neurons – Fixing the frequency







- Radio Frequency Emissions above threshold current
- Emitted power follows activation function:

$$\phi(\boldsymbol{v}) = \max(0, \boldsymbol{a} + \boldsymbol{v}'\boldsymbol{b})$$

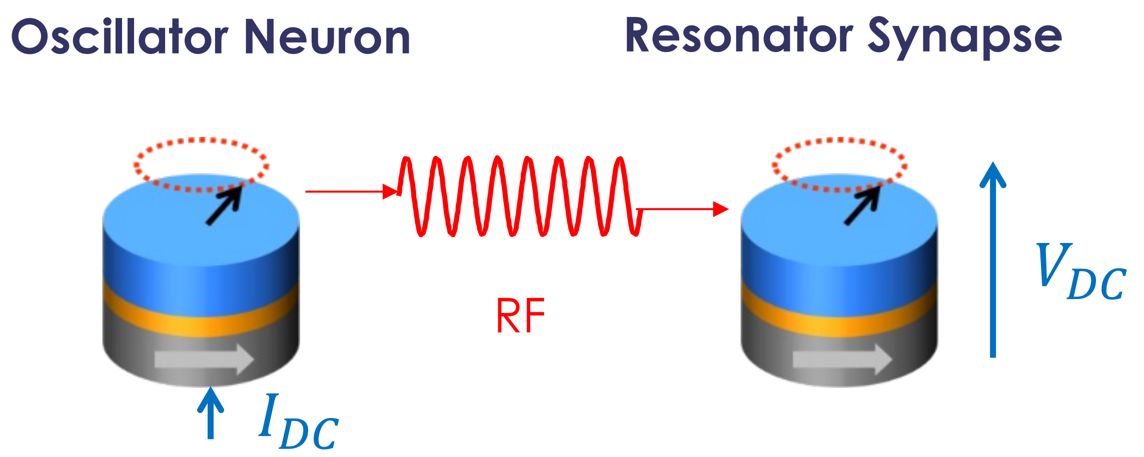
Today: current control of frequency

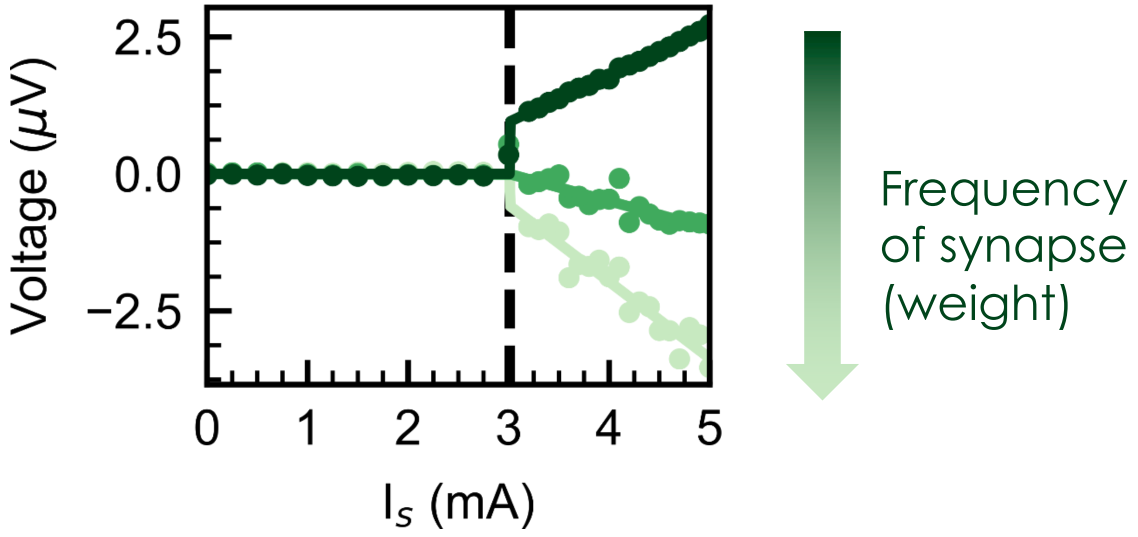
Future: oscillators with fixed frequency as in *B. Divinskiy et al., Nat. Commun.* ***10****, 1 (2019)*

THALES

Fully spintronic neural network

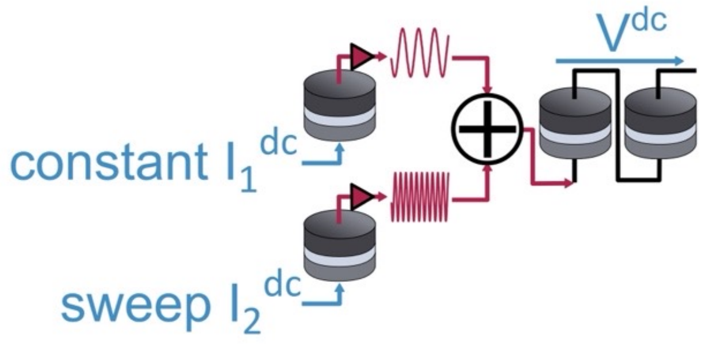
Connecting a nano-neuron to a nano-synapse

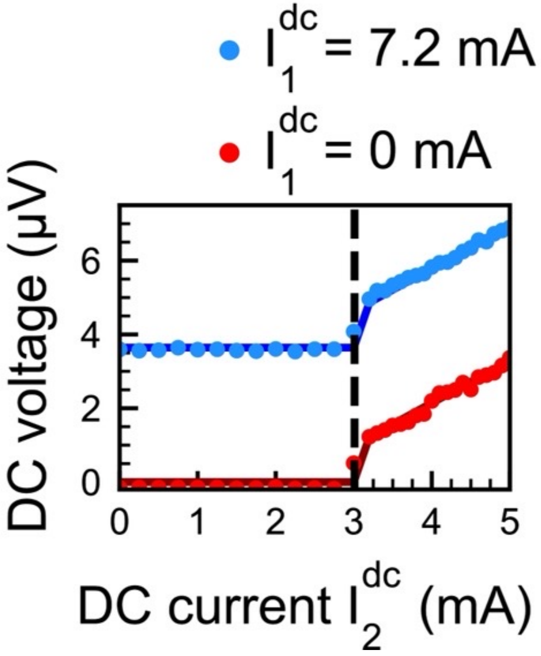


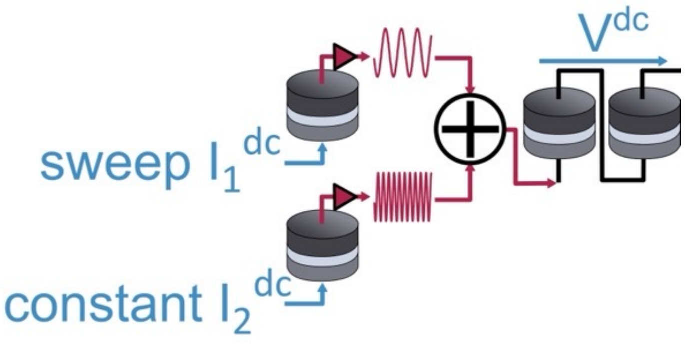


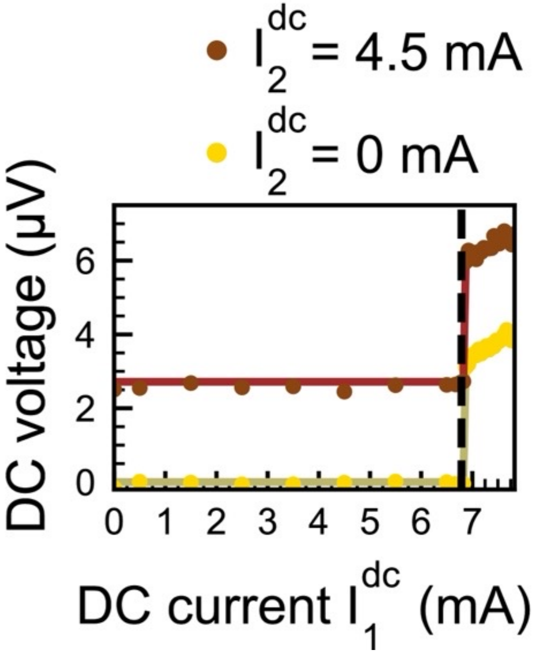
THALES

Connecting nano-neurons to nano-synapses







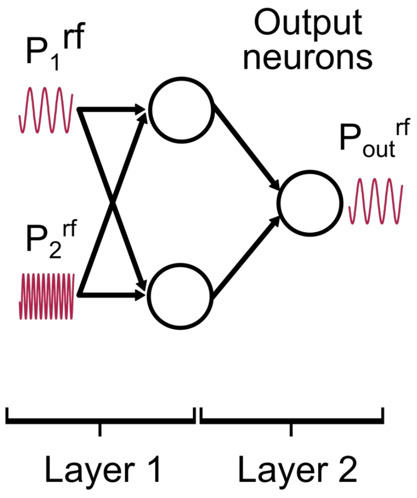


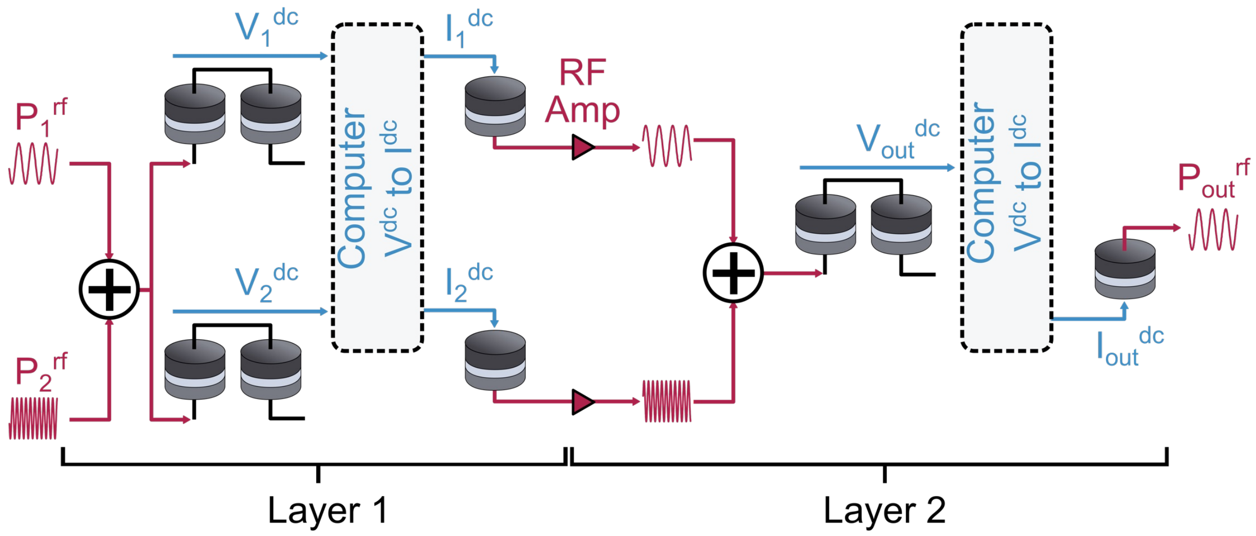
The weighted outputs of neuron 1 and 2 are summed

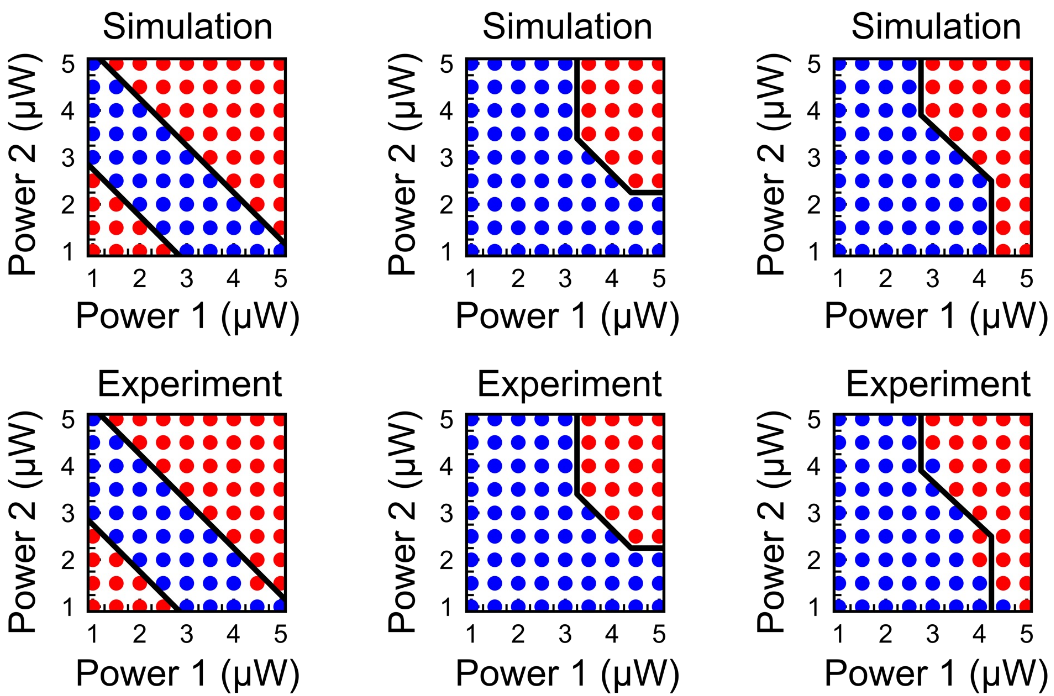
$$V = w_1 \times \phi_1(I_{s1}) + w_2 \times \phi_2(I_{s2})$$

THALES

Experimental non-linear classification







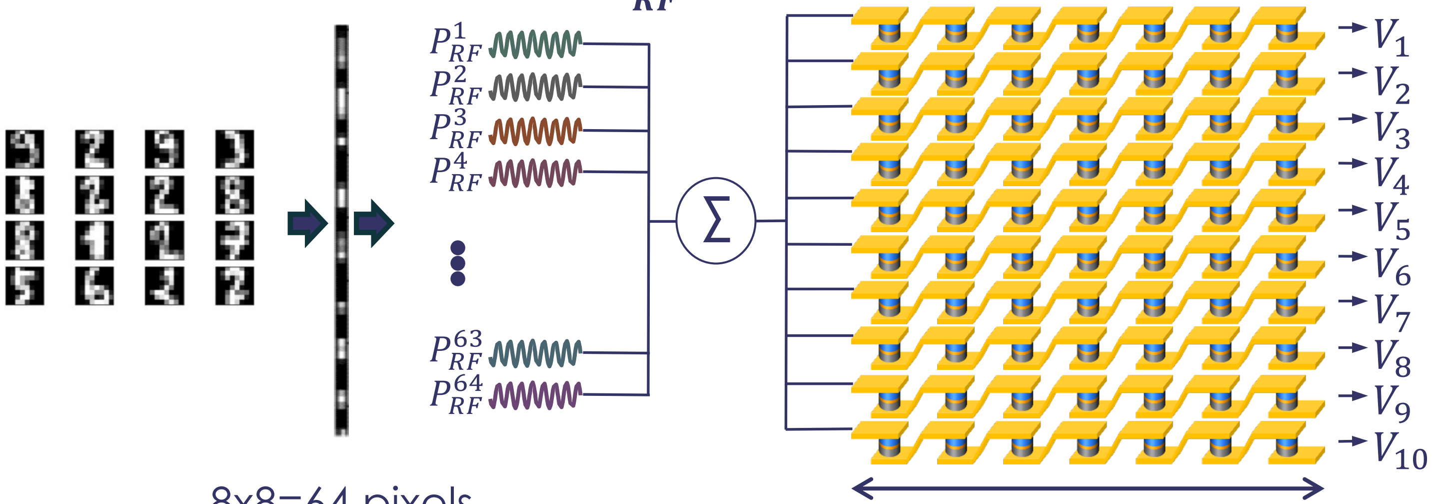
Each task: different set of weights

- $P_{out} = 0$
- $P_{out} \neq 0$

Large scale simulations

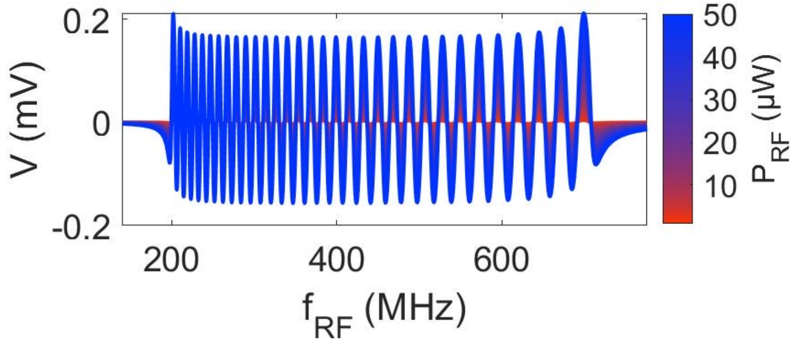
We simulate a larger network with physical models

Pixels converted to P^i_{RF}



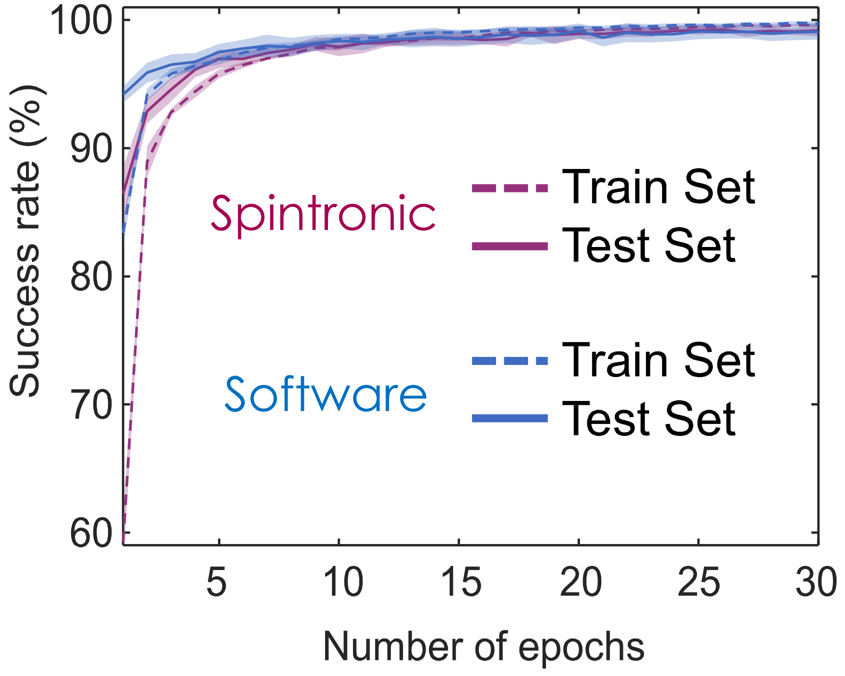
8x8=64 pixels
1797 pictures

64



PyTorch library
+
physical models

Precision > 99 %

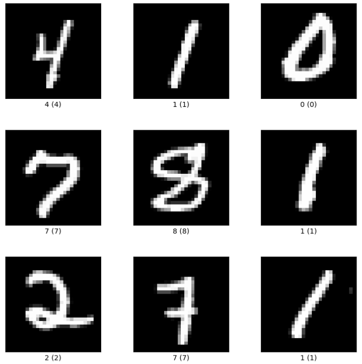


Leroux, Mizrahi et al., "Radio-Frequency Multiply-and-Accumulate Operations with Spintronic Synapses." *Physical Review Applied* 15, no. 3 (2021): 034067.

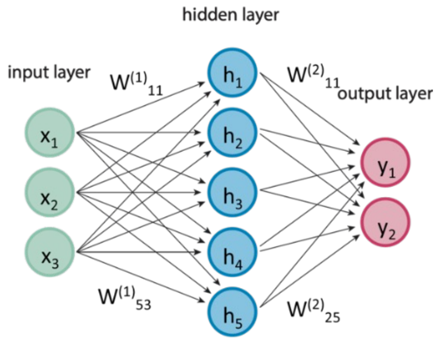
THALES

We simulate more complex architectures

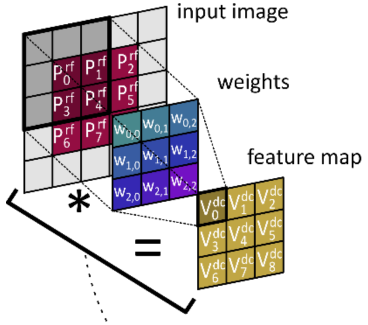
MNIST:
28 X 28 pixels



Multi-layer Perceptron

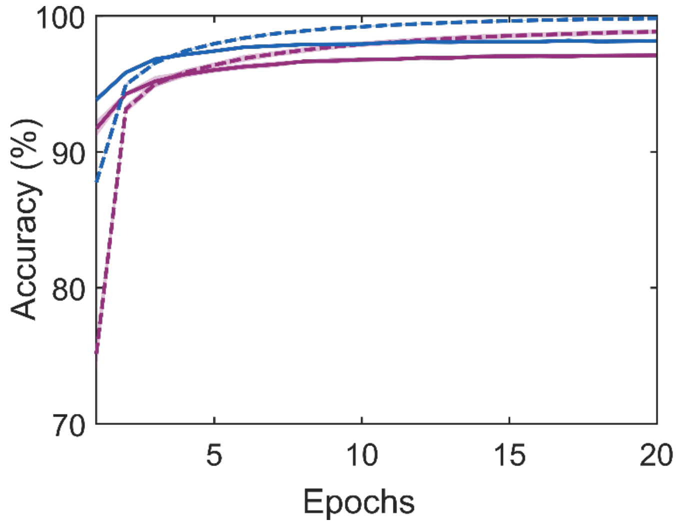


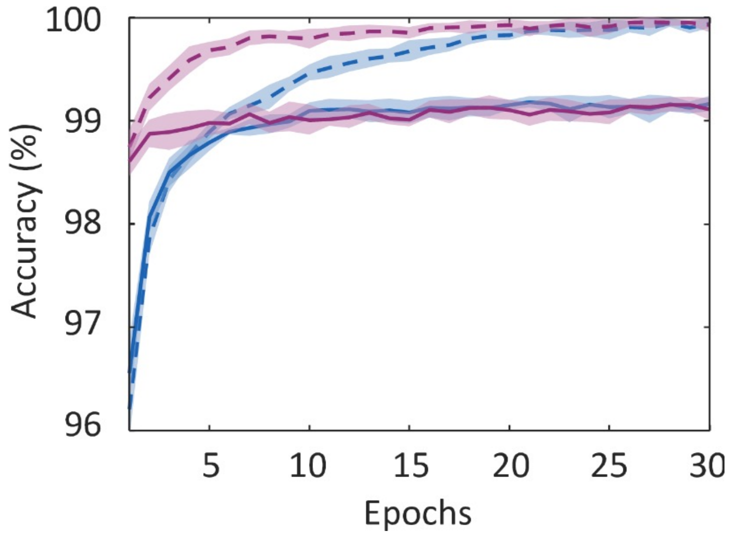
Convolutionnal Neural Network



Spintronic: Train Set (dashed), Test Set (solid)

Software: Train Set (dashed), Test Set (solid)

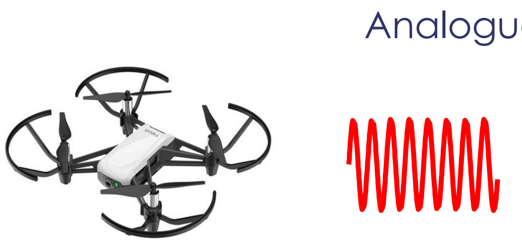


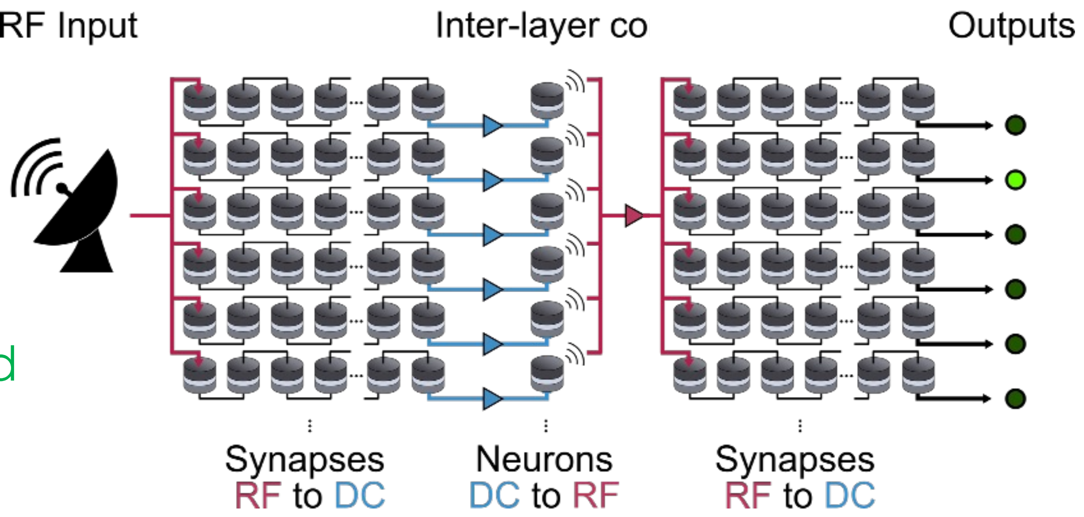


Leroux, Nathan, et al. "Convolutional Neural Networks with Radio-Frequency Spintronic Nano-Devices." *Neuromorph. Comput. Eng.* **2**, 034002 (2022) + Patent

THALES

Radiofrequency application: classify drones from their RF signals

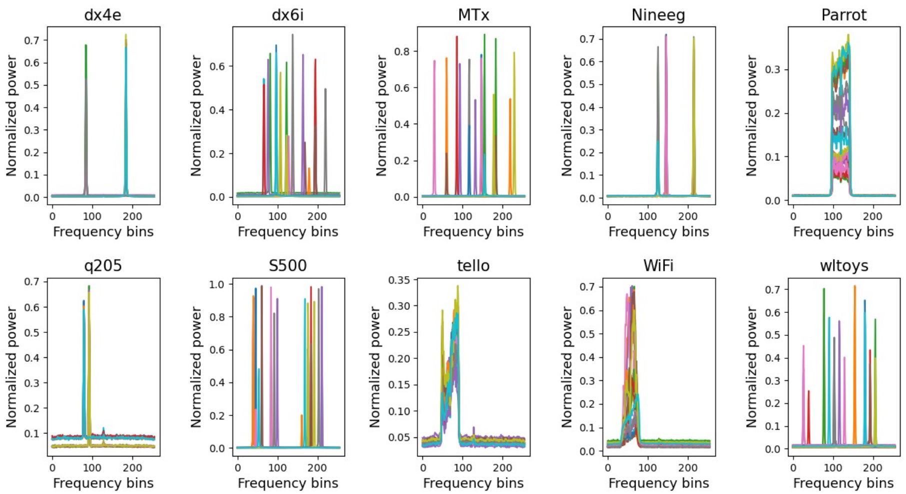


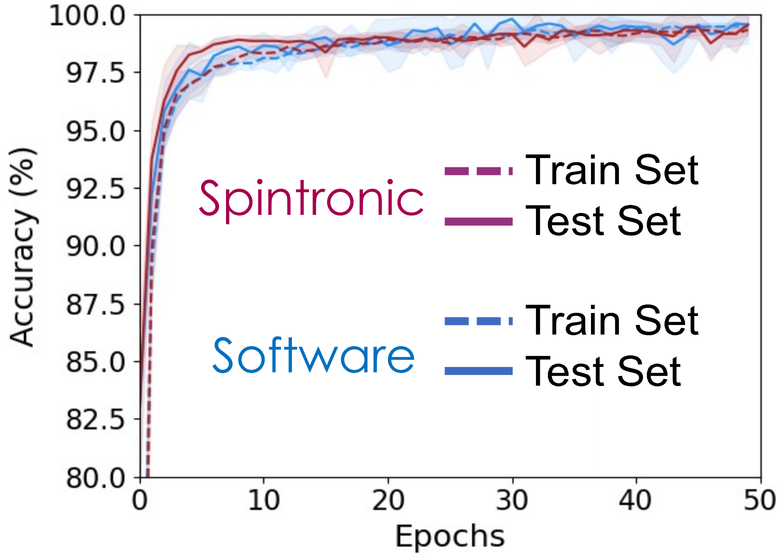


"This is a *Tello* drone"

No digitization required

10 classes of drones





Basak, et al. "Drone classification from rf fingerprints using deep residual nets." *2021 International Conference on COMmunication Systems & NETworkS (COMSNETS)*. IEEE, 2021.

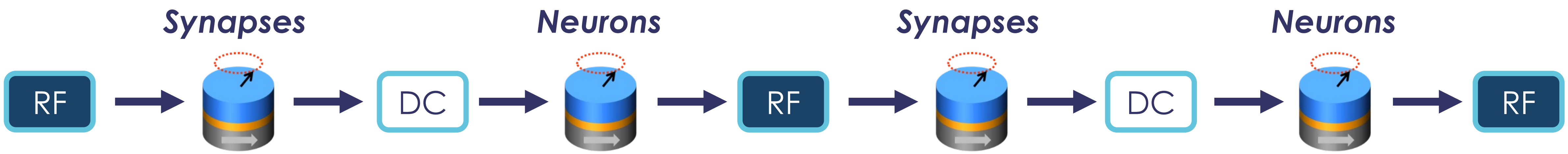
THALES

Comparison to other technologies

Spintronic RF-to-DC and DC-to-RF conversions enable a nice propagation of signals along the deep network

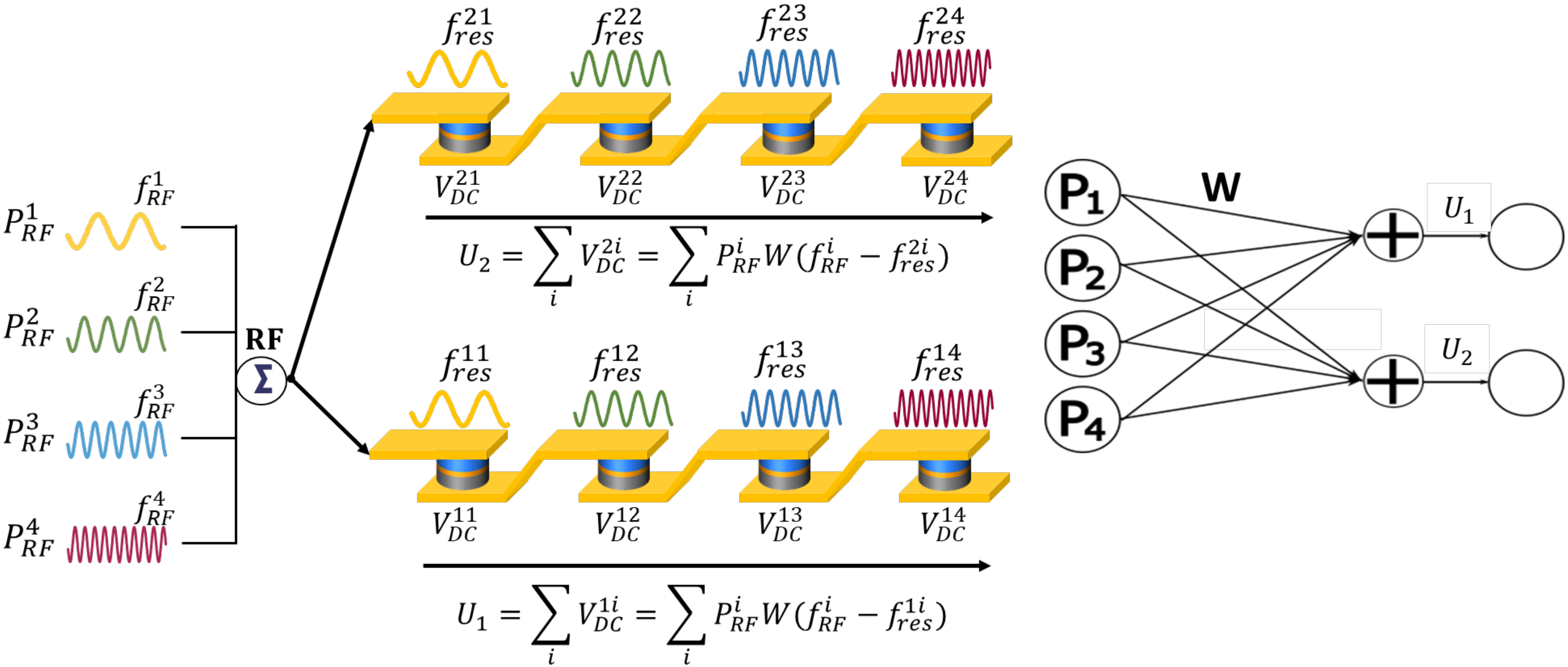
First fully nano multilayer neural network

→ *Nano-Synapses* → *Nano-Neurons* → *Nano-Synapses* → *Nano-Neurons* →



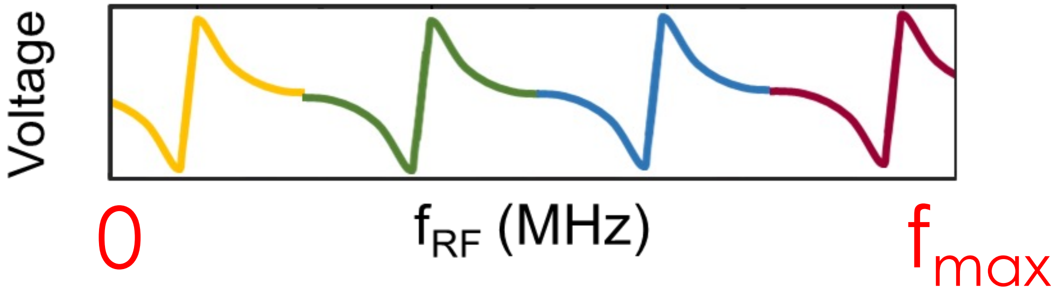
⇒Same device for neurons and synapses
⇒Stack layers alternating DC and RF for depth

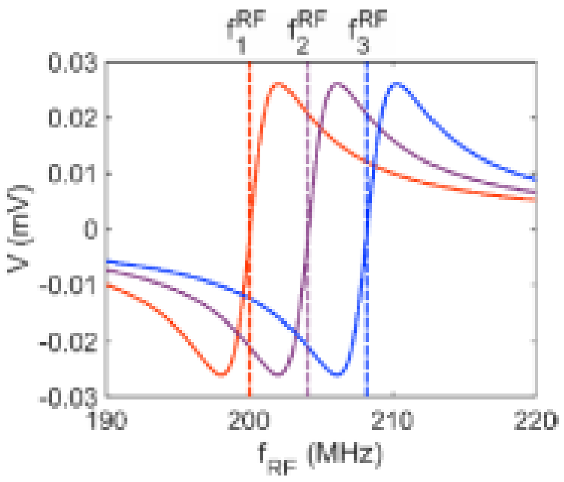
RF-to-DC connections implement a weighted sum with real activations and weights, without sneak paths

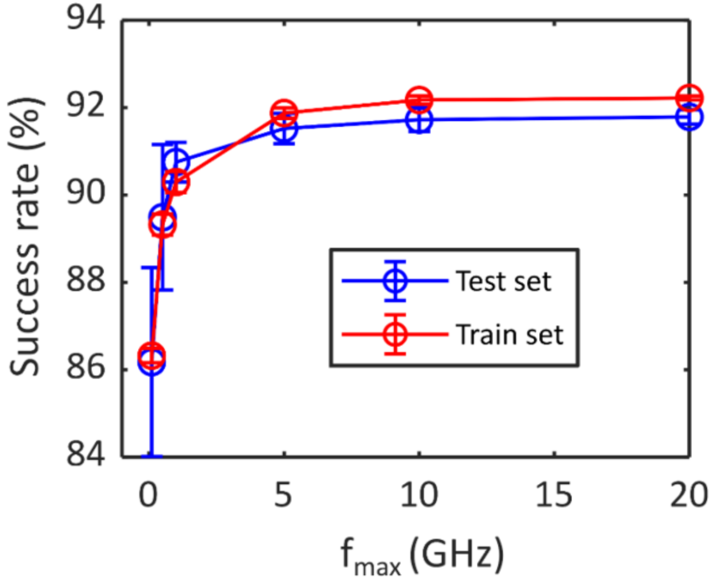


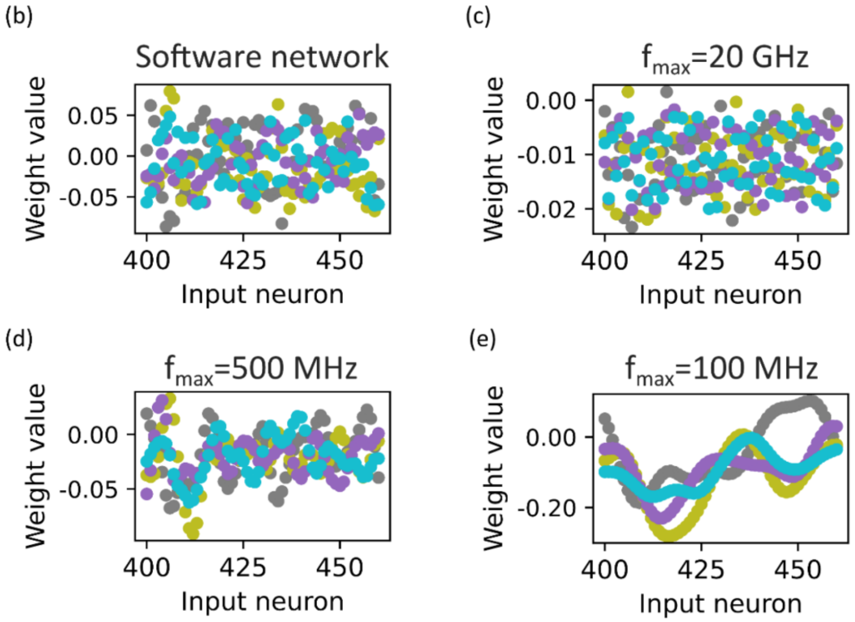
THALES

With a frequency range of 10 GHz, 784 neurons can be connected to 784 synapses



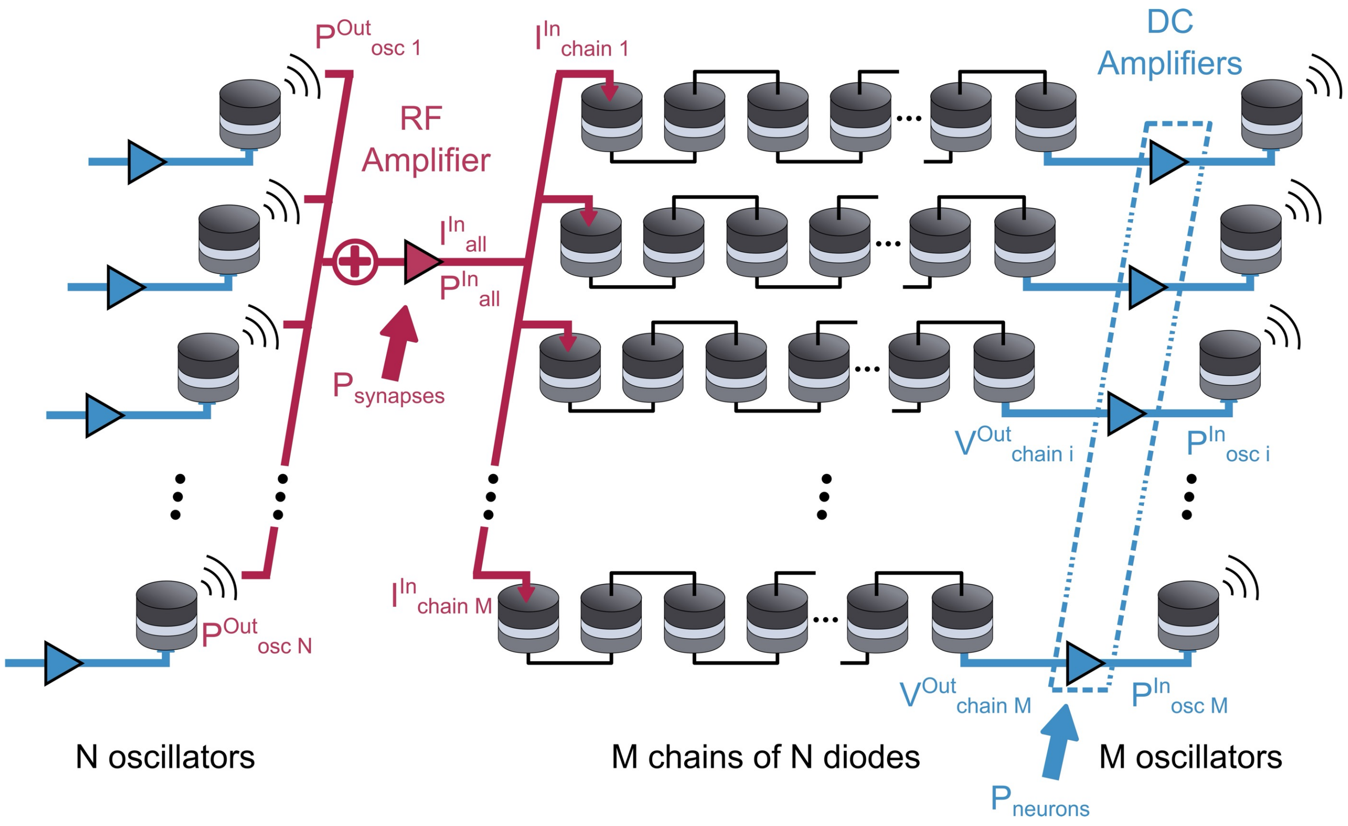






We take overlaps into account

10 fJ/synapse and 100 fJ/neuron for MTJs with 20 nm diameter



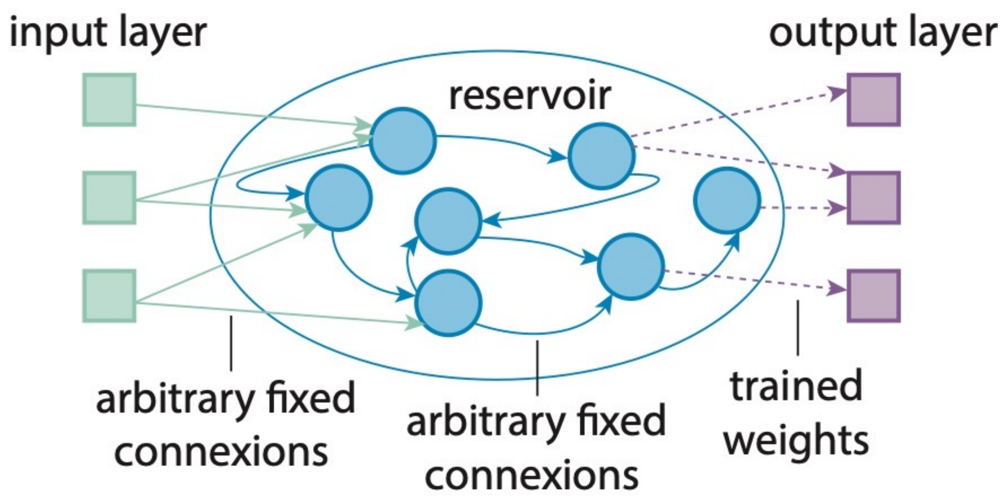
Comparable (better) to energy consumption estimations for memristive or optical devices

THALES

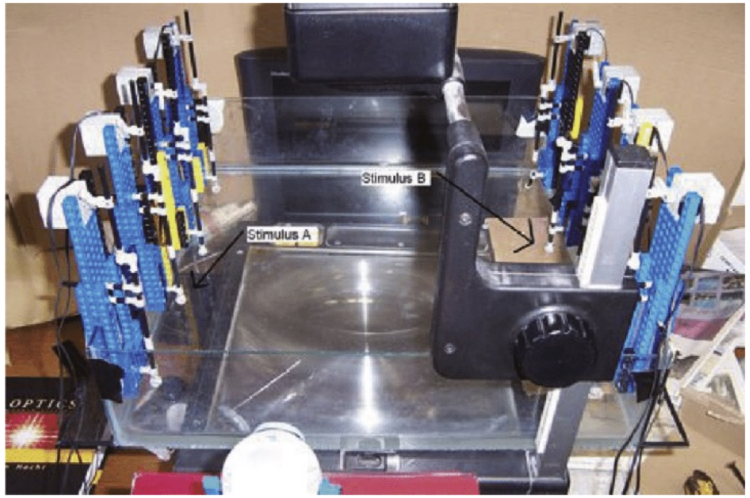
How to train your system?

Learning is necessary

It is possible to use the hardware for inference only and use a conventional computer for training but that limits what you can do with the hardware
(also might not be where emerging technologies are competitive)



NOT SCALABLE



Recent advances in physical reservoir computing: A review, 2019

Training the connections over several layers is required for complex tasks
Methods such as reservoir computing do not scale ☹

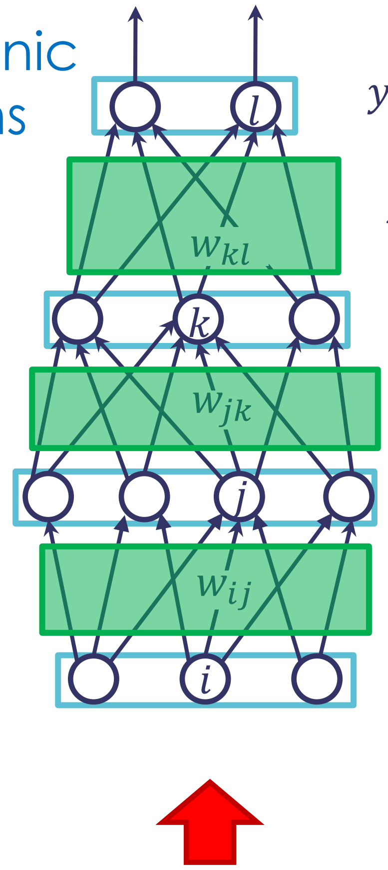
THALES

On-chip learning challenge : the backpropagation of errors is not hardware friendly

The hardware neural network is used for the forward pass

Electronic neurons

Electronic synapses



$$y_l = \rho(z_l)$$

$$z_l = \sum w_{kl} y_k$$

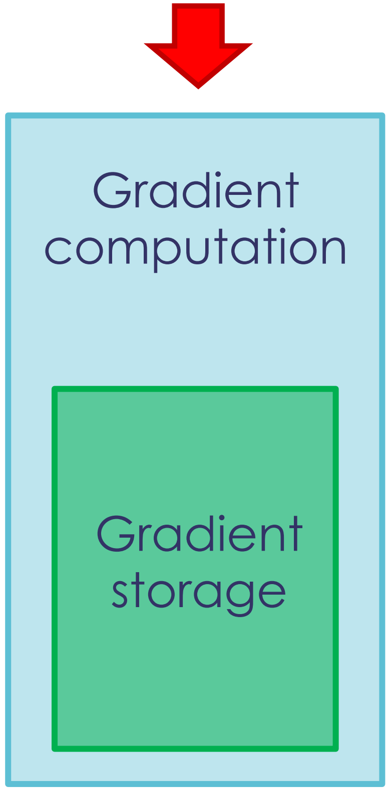
$$y_k = \rho(z_k)$$

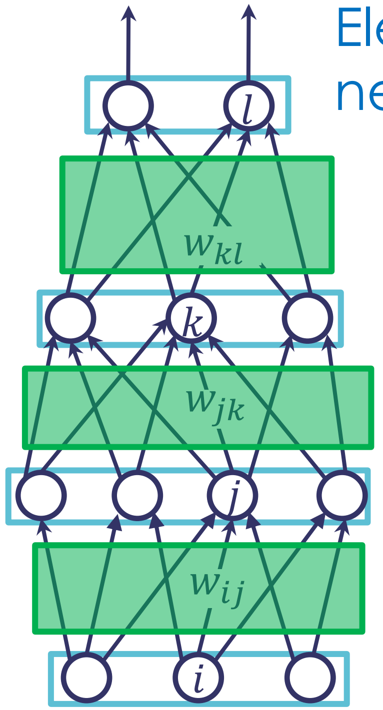
$$z_k = \sum w_{jk} y_j$$

$$y_j = \rho(z_i)$$

$$z_j = \sum w_{ij} y_i$$

A second circuit to compute and store gradients in the backward pass





Electronic neurons

Electronic synapses

Cost function: $C = \frac{1}{2}(y_l - t_l)^2$

$$\frac{\partial C}{\partial y_l} = y_l - t_l$$

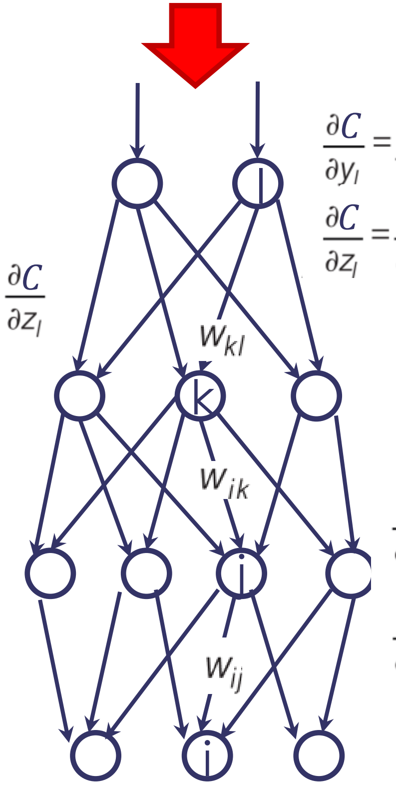
$$\frac{\partial C}{\partial z_l} = \frac{\partial C}{\partial y_l}\frac{\partial y_l}{\partial z_l}$$

$$\frac{\partial C}{\partial y_k} = \sum_{l\,\varepsilon\,\text{out}} w_{kl}\frac{\partial C}{\partial z_l}$$

$$\frac{\partial C}{\partial z_k} = \frac{\partial C}{\partial y_k}\frac{\partial y_k}{\partial z_k}$$

$$\frac{\partial C}{\partial y_j} = \sum_{k\,\varepsilon\,\text{H2}} w_{jk}\frac{\partial C}{\partial z_k}$$

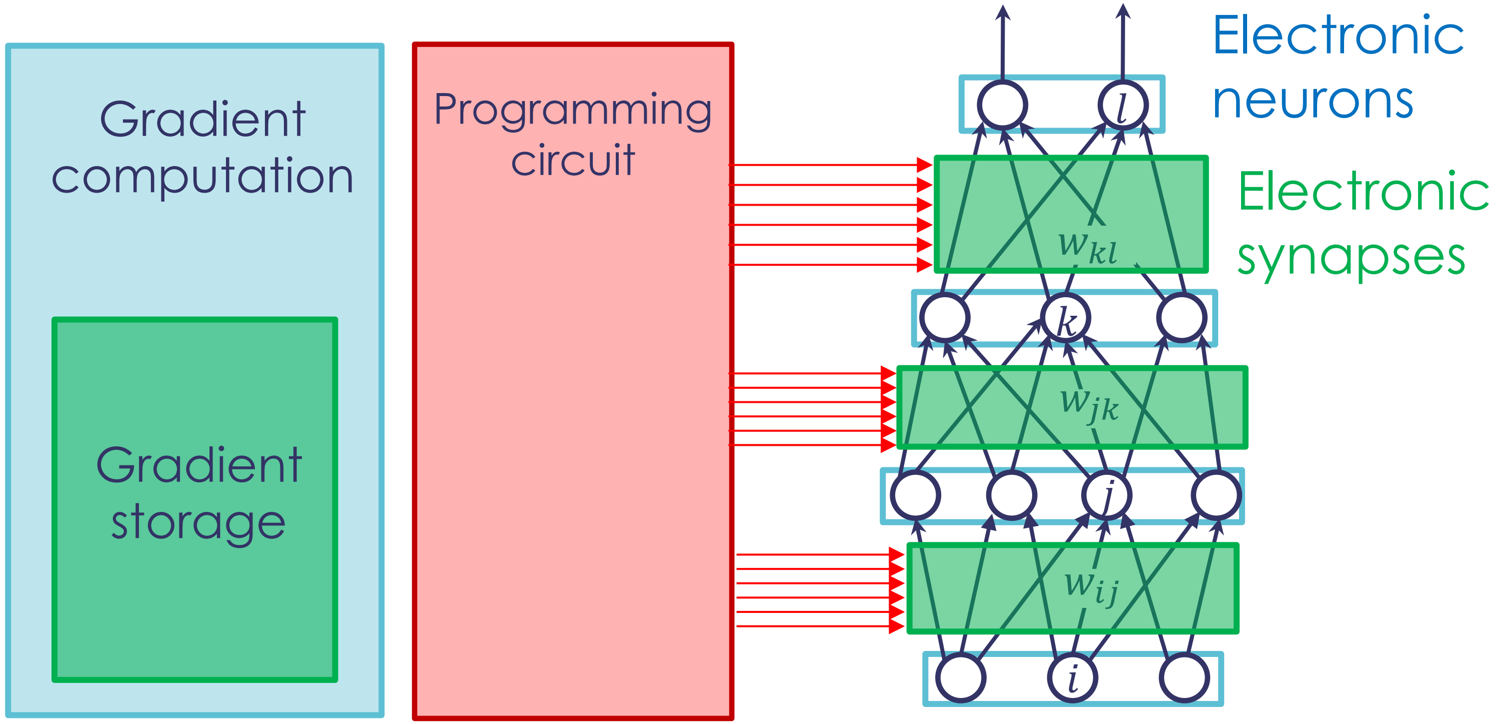
$$\frac{\partial C}{\partial z_j} = \frac{\partial C}{\partial y_j}\frac{\partial y_j}{\partial z_j}$$

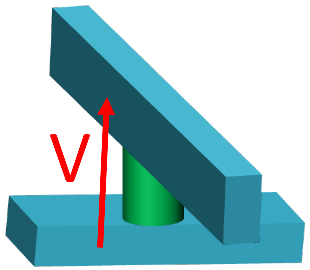


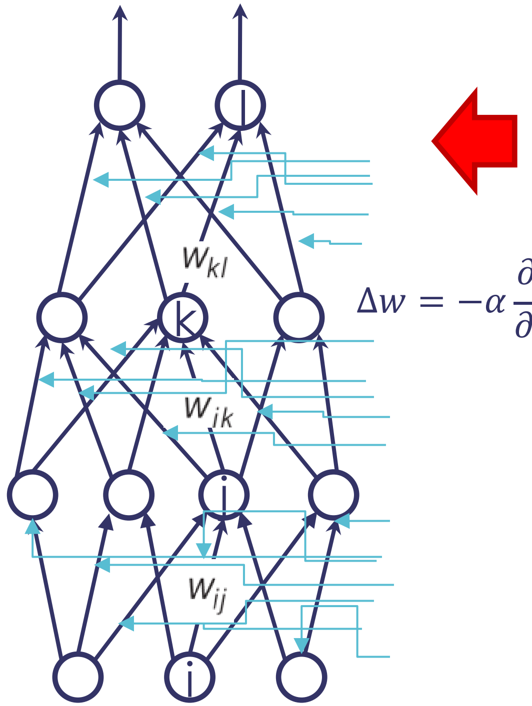
$$\Delta w_{ij} \propto -\frac{\partial C}{\partial w_{ij}}$$

A third circuit to program the synapses to the new weight values

$$\frac{\Delta w}{w} < 10^{-5}$$





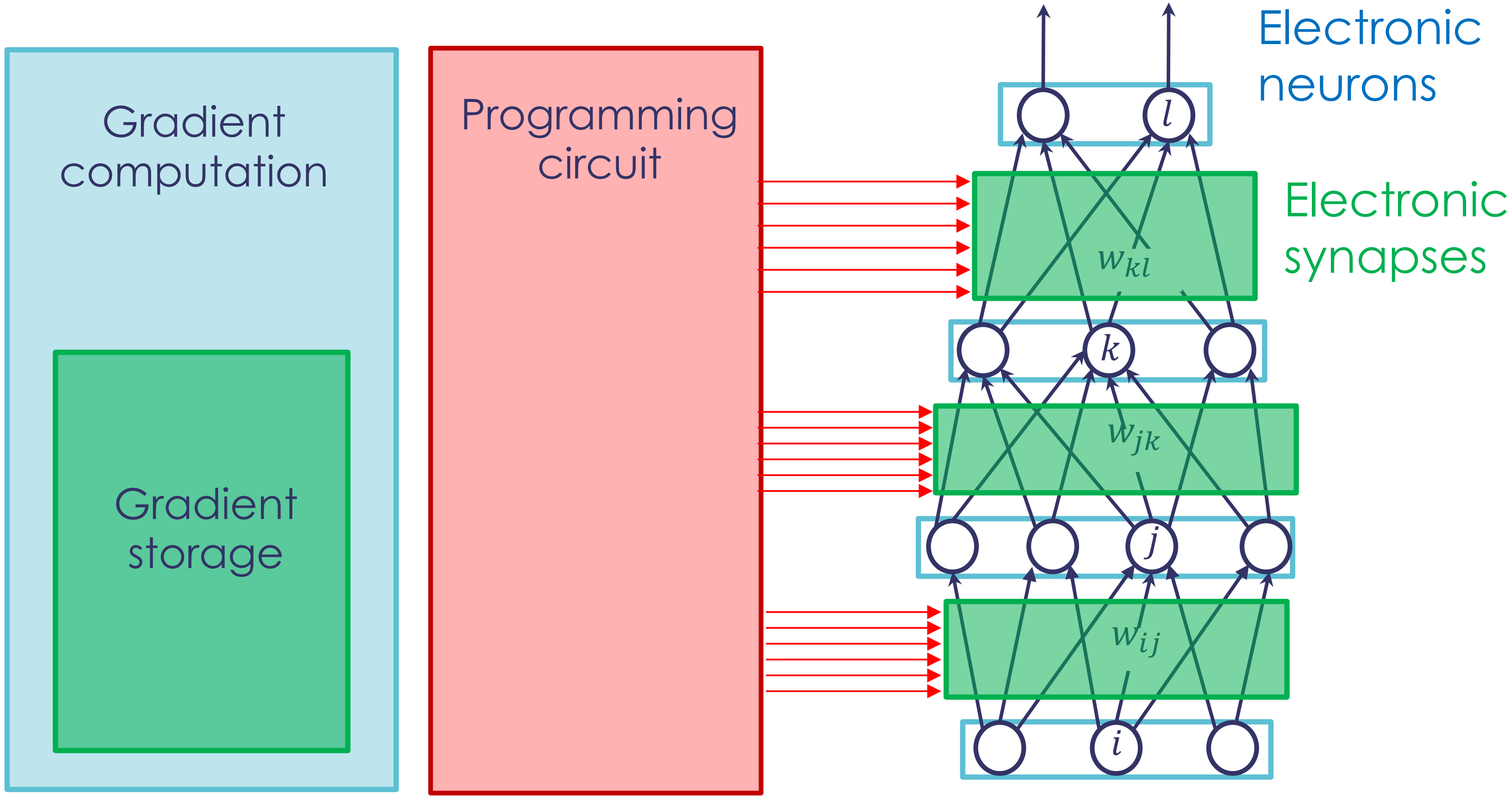


$$\Delta w = -\alpha \frac{\partial C}{\partial w}$$

40

THALES

Building low power, compact circuits to implement backpropagation on chip is a tremendous challenge



THALES

Backpropagation vs. neuro/physics inspired rules

Backpropagation is state of the art performance on actual tasks

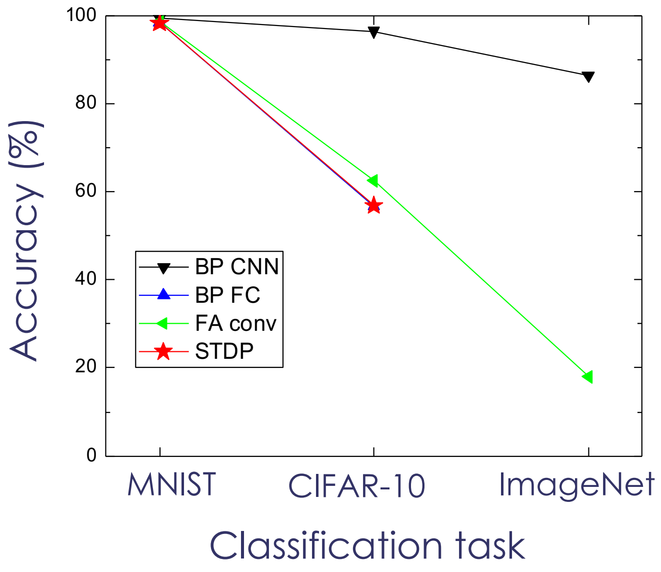
But… Not hardware friendly

Neuro/physics rules are hardware friendly:
Local (synapse modified only by neurons around)
Self-learning by the physics of system

But…. Performance on hard tasks is low because they do not minimize the global error

Can we merge the two to get the advantages of both?

Does the brain perform some kind of backpropagation?

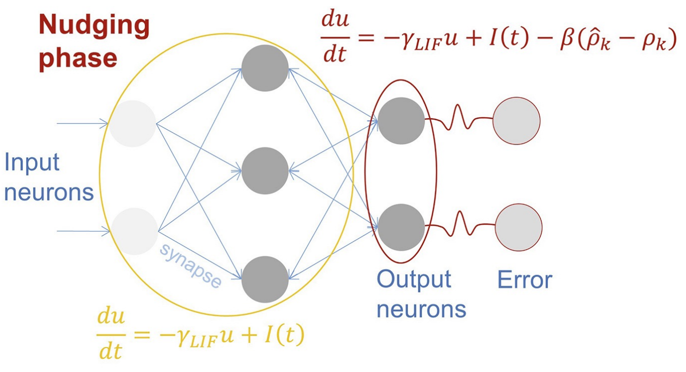


Lillicrap, T.P., Santoro, A., Marris, L. *et al.* Backpropagation and the brain. *Nat Rev Neurosci* **21**, 335–346 (2020)

Video of Hinton « Stanford Seminar - Can the brain do back-propagation? »
https://www.youtube.com/watch?v=VIRCybGgHts

Towards self-learning with physical systems

- Hot topic in machine learning / computational neuroscience: find algorithms where dynamics of system computes approximations of gradients of errors
- And the dynamics of the system should also update the synaptic weights (self learning)



Martin, et al. "Eqspike: spike-driven equilibrium propagation for neuromorphic implementations." Iscience 24.3 (2021)

Tough problem but exciting
Opportunities for physicists
=> Really take advantage of rich physics of spintronics devices

Inspirations: neuroscience, wave physics, minimizing the energy of a system etc.

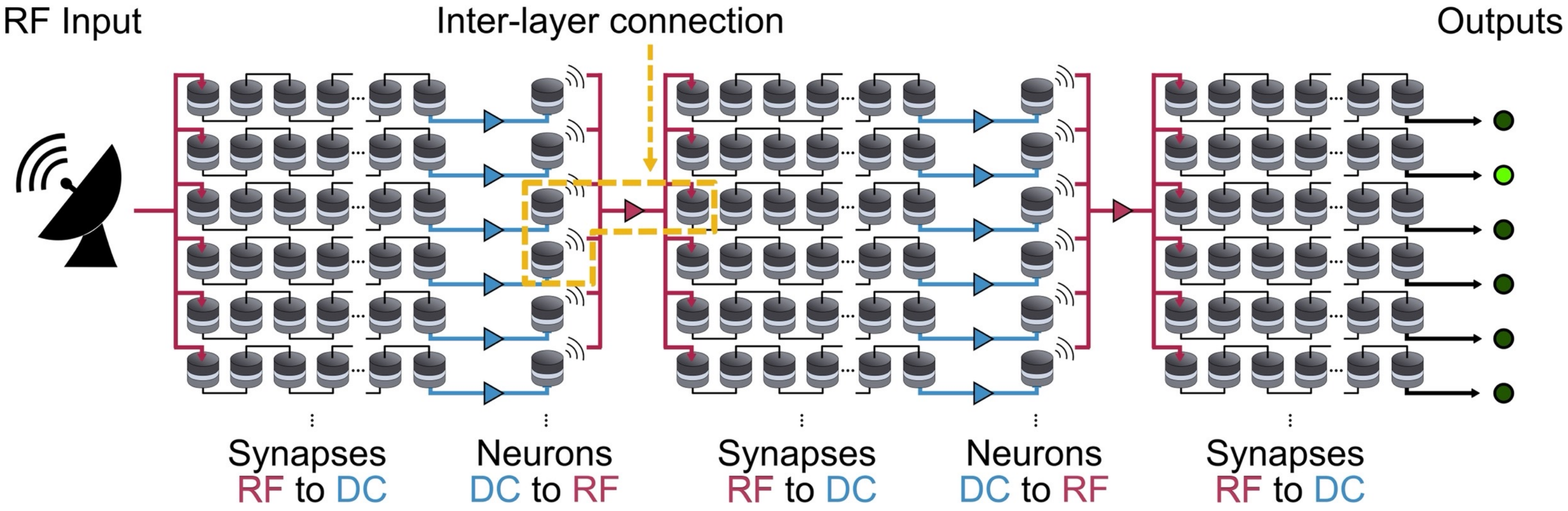
Scellier, Benjamin, and Yoshua Bengio. "Equilibrium propagation: Bridging the gap between energy-based models and backpropagation." *Frontiers in computational neuroscience* 11 (2017)
Lopez-Pastor, Victor, and Florian Marquardt. "Self-learning machines based on Hamiltonian echo backpropagation." *arXiv:2103.04992* (2021).

Towards deep RF spintronic neural networks

- **First experimental** demonstration of **communication** between **multiple layers** using nano-devices
- **Non-linear classification** of two inputs with **high accuracy**
- **Simulations of large networks** on benchmark tasks and RF applications
- Next steps:

  => **scale** up network in **CMOS/MTJ chip** → potentially hundreds of connections per chain

  => implement **self-learning** using the **dynamics** of the devices



We are hiring PhD students and postdocs!

Theory weighted sum: N. Leroux, A. Mizrahi *et al. Phys. Rev. Appl.* **15**, 034067 (2021)
Experiment weighted sum: N. Leroux, A. Mizrahi *et al*., Neuromorph. Comput. Eng. **1**, 011001 (2021)
CNN: N. Leroux *et al Neuromorph. Comput. Eng*. **2,** 034002 (2022)
Experiment Full network: *Manuscript in Preparation*