Challenges for Neuromorphic Circuits

Damien Querlioz

Institut d’Electronique Fondamentale

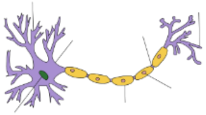
Univ. Paris-Sud, CNRS

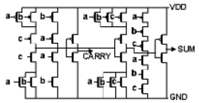
Université Paris-Saclay

I want to design a hardware that works « like the brain »

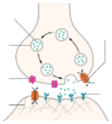
- ❑ Here we do not leverage existing hardware
- ❑ We design a **specific hardware**, from transistor level, hoping maximum energy efficiency
- ❑ **What are the challenges?**

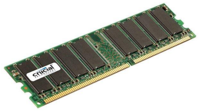
Three big challenges



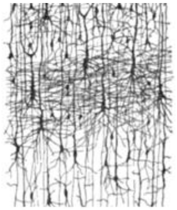


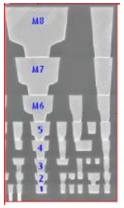
- ❑ Computing





- ❑ Memory



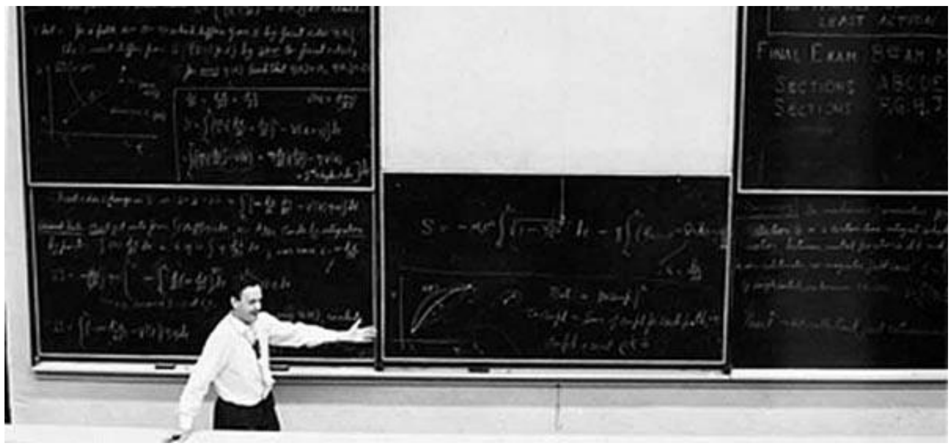


- ❑ Communication

Let's compare the brain and electronics…

1. Computing

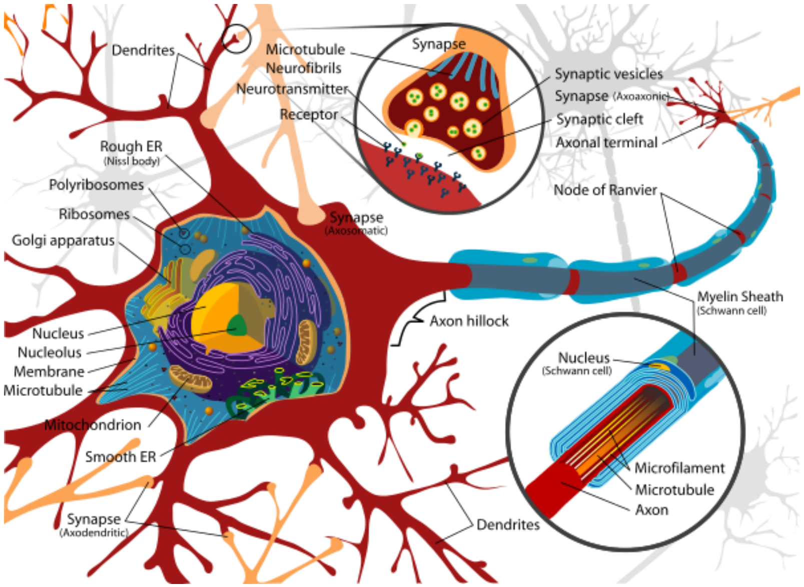
- **The essence of my system: computing stuff**



- How do I design my basic computing units?

Biology: Neurons and Synapses

Actual neurons and synapses



wikimedia

Complex structure

Hundreds of ion pumps and ion channels

Each element (synapse, ion channel, compartment…) has supercomplex nonlinear dynamics

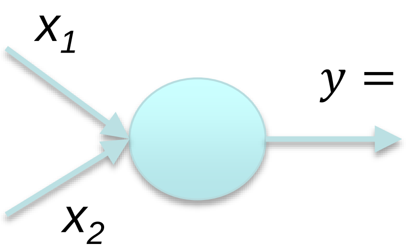
…

- We are going to design a system w. neurons and synapses
- But do I want all this complexity?

Which neuronal model?

The most abstracted neuronal models

State neuron

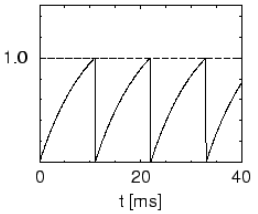


$$y = f\left(\sum_i w_i x_i\right)$$

Leaky integrate and fire neuron

$$\frac{dX}{dt} + gX = input$$

If $X > X_{th}$ declare a spike



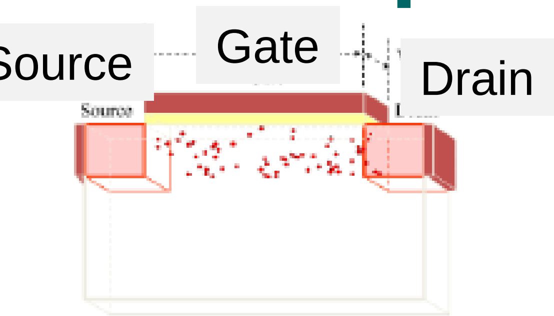
❑ Between the two extremes, which level of abstraction do I want?

And how do I fabricate my neurons?

- ❑ No ions un microelectronics technology!

- ❑ How do I design highly energy efficient neurons and synapses?

A very bioinspired aproach: computing w. transistors leakage

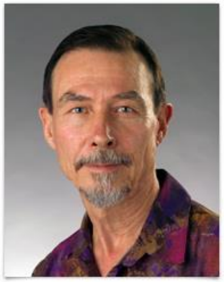


$$I_D \propto e^{\frac{\kappa}{U_T}(V_G - V_S)} \quad *$$

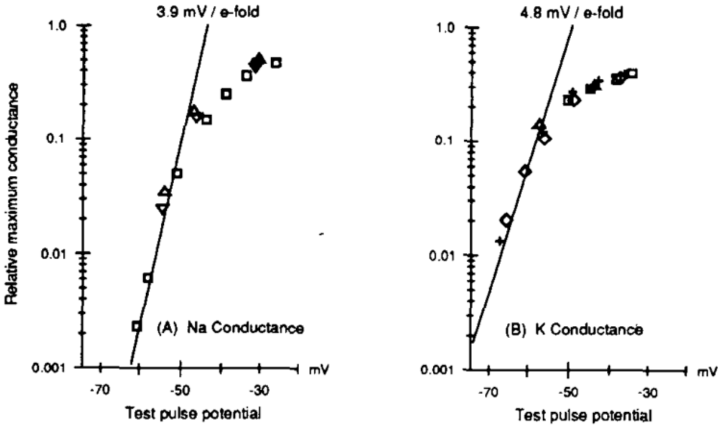
Transistor in the OFF state

* Fineprint: it's a little more complicated than that…

- ❑ Transistor leakage physics is well known. Normally a problem!
- ❑ But, inspired by biology, we can use it to compute



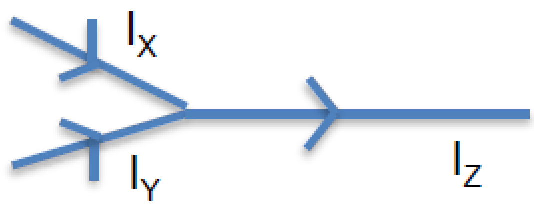
Carver Mead



Biology is using **exponential I-V curve** for computation

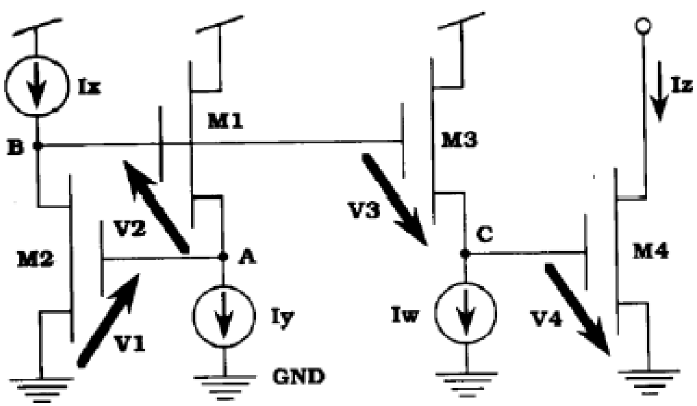
Compute with leakage

- Current mode. Low current / as slow as te brain
- Arithmetic operations easy with Kirchoff law!!!
- Sum or difference



$I_Z = I_X + I_Y$

- Product/ratio
  « translinear loop »



$V_1 + V_2 + V_3 + V_4 = 0$

$I_D \propto e^{\frac{\kappa}{U_T}(V_G - V_S)}$

$$I_Z = \frac{I_X I_Y}{I_W}$$

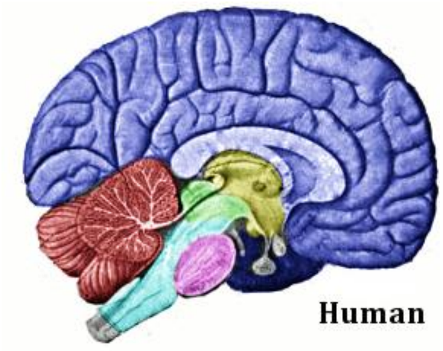
Also differential equations

- A circuit to solve $dX/dt+gX=gE$ (e.g. for leaky integrate and fire circuit)

(on paperboard)

- Power consumption for « processing » 1 spike
- **Few pA*0.1s*1V**
- pJ / operation!

How about human brain?



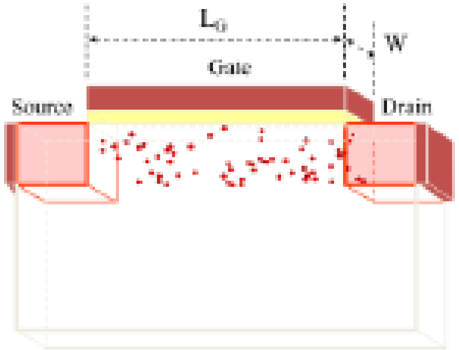
- 20W
- 10^{11} neurons, 10^{15} synapses
- 10 event/neuron/s

- **My guess** 200pJ/spike or 20fJ/synaptic event…

BUT drawback…

- Let's go back to our differential equation…

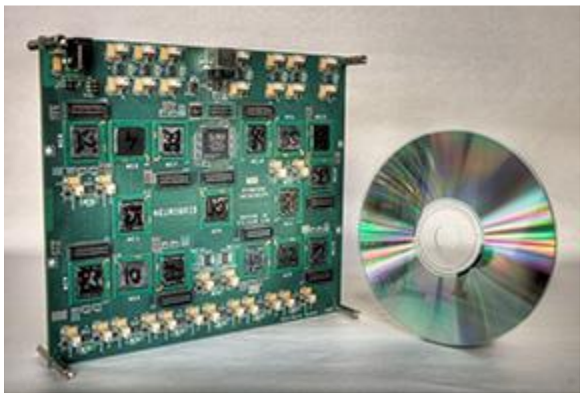
- In reality each transistor has different number of dopants, and each I_0 is different



- **Each neuron will be different!**

And computing is not everything

- Stanford Univ's beautiful Neurogrid system



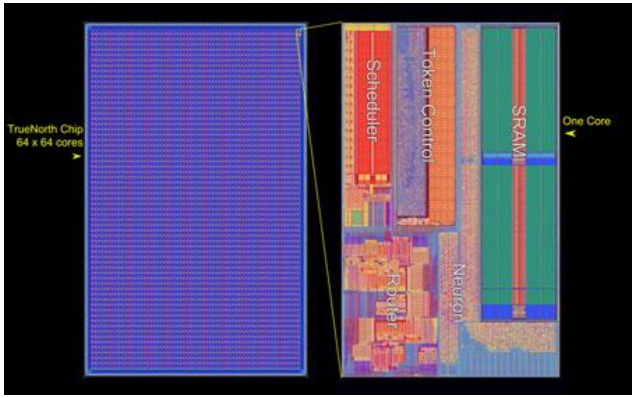
Benjamin et al, Proc IEEE 2014

1 Million neurons with associated memory and communication
Neurons are pure subthreshold CMOS

- Incl. memory and communication: Stanford Neurogrid
  1nJ/operation, not pJ!!!

And if we do it w. digital logic?

- ❑ IBM: Truenorth
  special 28nm process with ultralow leakage
- ❑ Digital neurons « quite » big: 1300 gates (1000 for neuron + 300 for random number generation)
- ❑ But very fast -> multiplexing

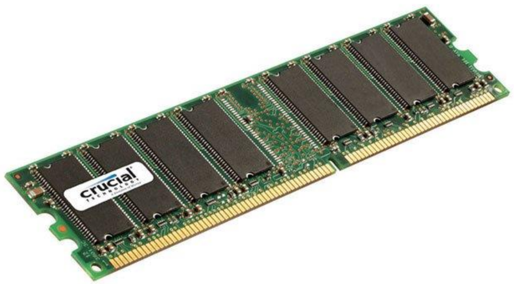


Merolla et al, Science 2014

- ❑ Everything on chip
- ❑ Power consumption only 26pJ/operation!!!

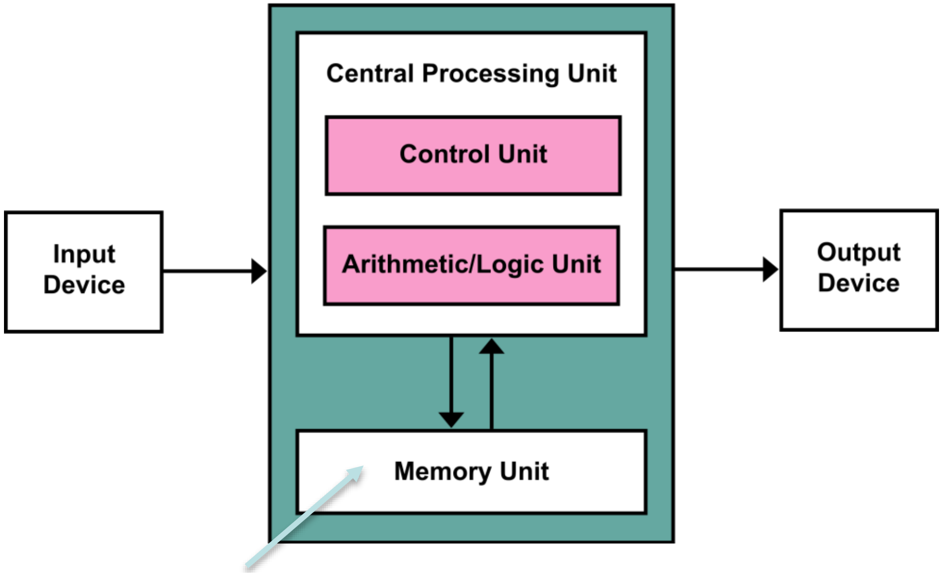
2. Memory

- Memory: compare computer and the brain



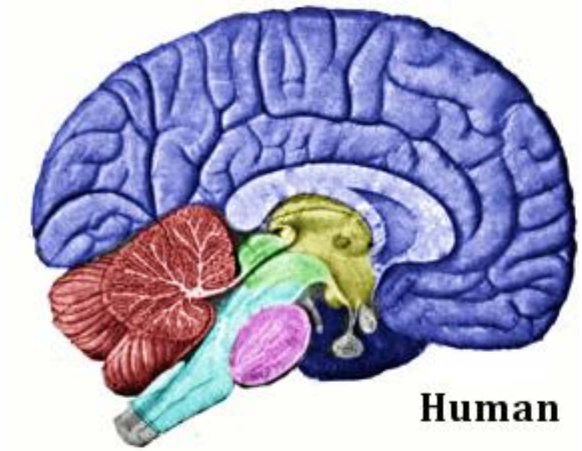
Computer vs. brain

Computer: von Neumann architecture



*PROGRAM **AND** DATA*

Brain



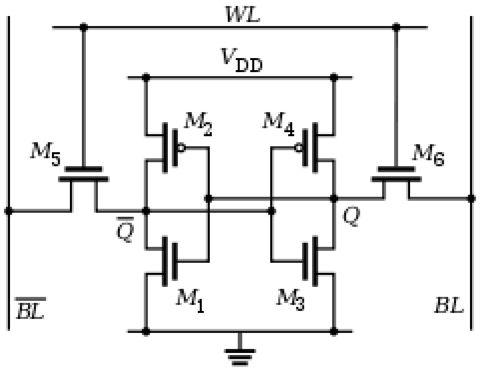
« Program » memory: everywhere!!!
« Data » memory: some areas a little specialized, but nevertheless a little everywhere

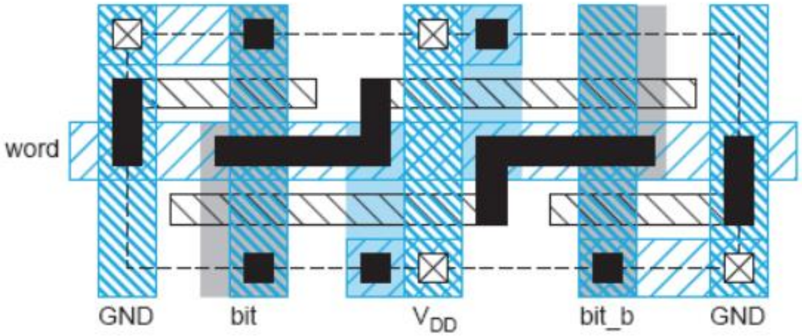
Distributed memory for neuromorhic

- Neuromorphic systems need a lot of memory
  - Synaptic weight
  - Network topology (usually not entirely hardwired)
  - Neuron parameters
- Distributed memory, like in the brain, is better
- Sometimes, neuromorphic systems look like big memory chips!!

The issue: the SRAM cell

- In CMOS, distributed memory is HUGE!!!
- 1 bit SRAM cell ~140F²

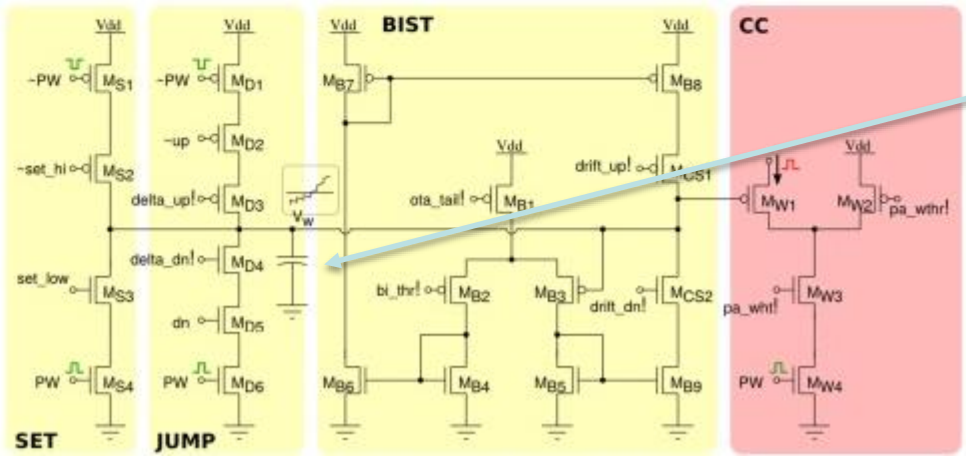




- Also, it is **volatile**

How about learning?

- Learning in CMOS: distributed memory for synaptic weight + learning circuit



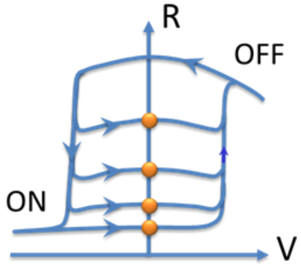
ENORMOUS
Many pF

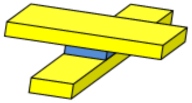
STDP circuit
Indiveri, Front Neurosci 2015

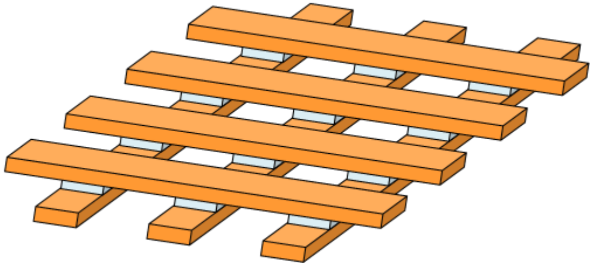
- This is HUGE . You cannot have many synapses

RRAM/memristors to the rescue?

- RRAMs / memristors are
  - Compact
  - Nonvolatile
  - Intrinsic learning capability





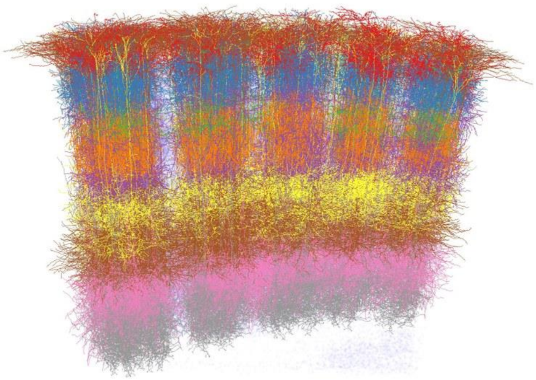


Courtesy of J. Grollier

- Ideal solution for memory in neuromorphic circuits?

3. Communication

- Brains: a neuron receives inputs from 10k other neurons…



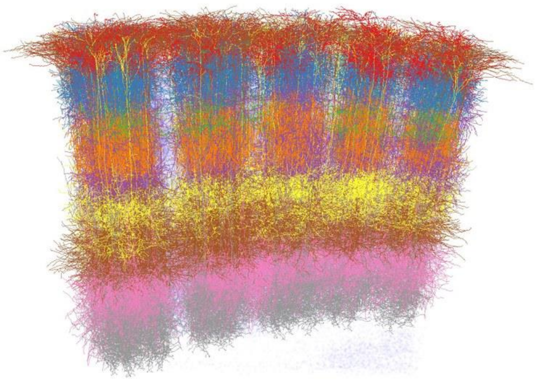
Rat cortex
Oberlaender et al

- Probably key element of brains’ capabilities

- Can we replicate that?

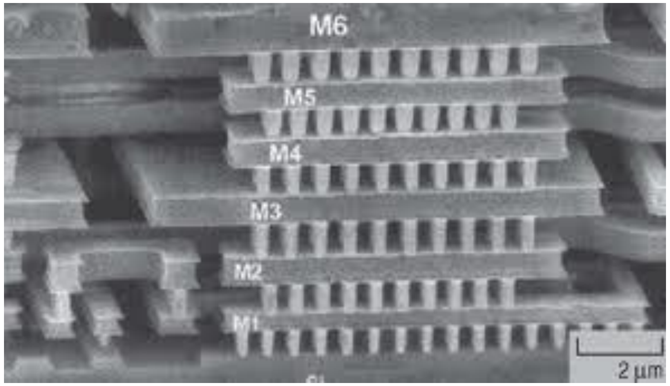
Brain vs. microelecronics

Brain: complex 3-D structure



Rat cortex
Oberlaender et al

Microelectronics: very planar interconnection
Hard to wire stuff!!!

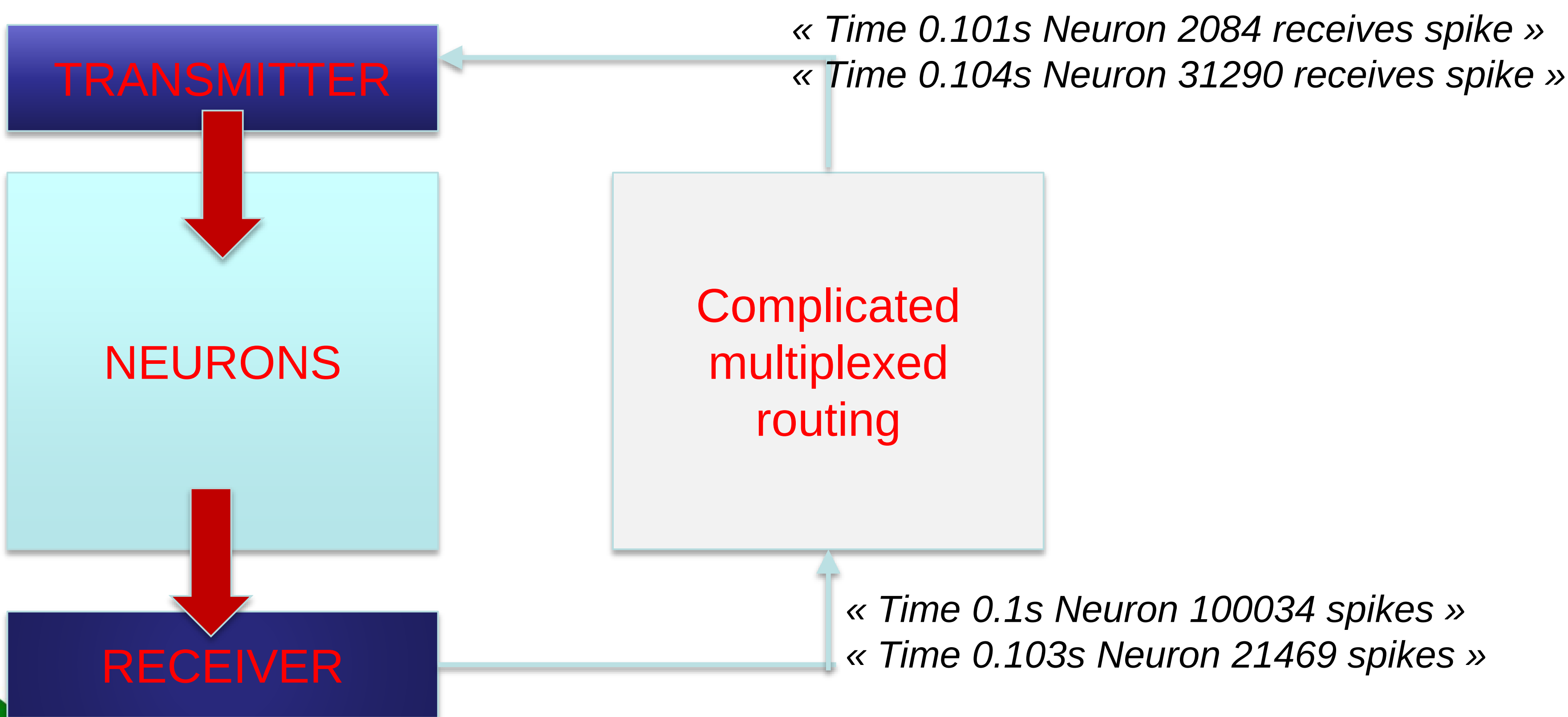


UMC, TW

- In microelectronics, we cannot wire like the brain

Connecting neurons: multiplexing!

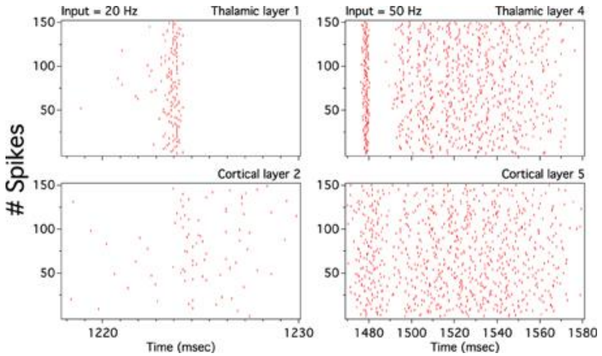
- BUT brain is slow and microelectronics is fast…



- NOT Bioinspired! Leverages CMOS speed!

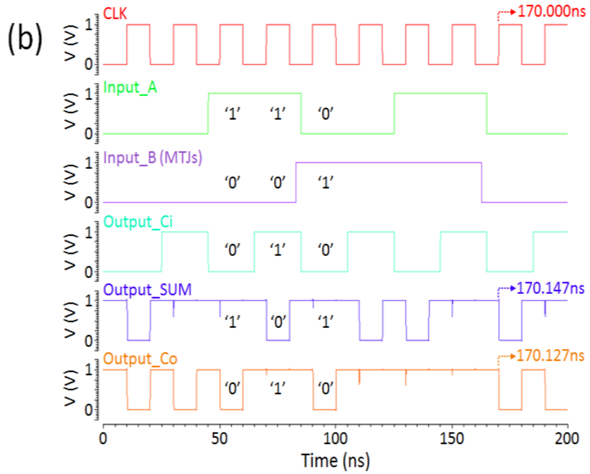
How to synchronize all these guys?

- BRAIN: asynchronous



Some random neuroscience paper

- Computers: synchronous (clocked)



Some random digital design paper

Asynchrony

- In neuromorphic design, many people actually do **asynchronous** chips

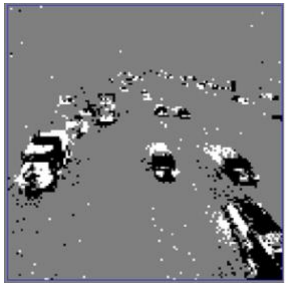
- To help you with design
  - Synchonous: EDA tools, multibillion $ industry
  - Asynchronous: …

- Why is asynchronous hard?
  - Dealing with delays
  - Example of 4-phase handshaking (on paperboard)

A solution for communication: Address Event Representation (AER)

- For asynchronous multiplexed communication
  - Asynchrnonous events with the associated address

- Handshaked



DVS camera (Tobi Delbruck, ETH Zurich) can communicate w. all kind of neuromorphic systems w. AER

Conclusion

- There are many technical options for neuromorphic, all with advantages/drawbacks. So really no universal solution. Technical choices will depend on
  - Speed needed (real time or accelerated)
  - Network size
  - Do I need learning?
- What complexity for neurons/synapse models ?

- Dedicated hardware. Very attractive but to what extent essential?

Thank you for you attention!