

Analogue building blocks for neural-inspired circuits

Steve Hall

Dept. Electrical Engineering & Electronics

s.hall@liv.ac.uk

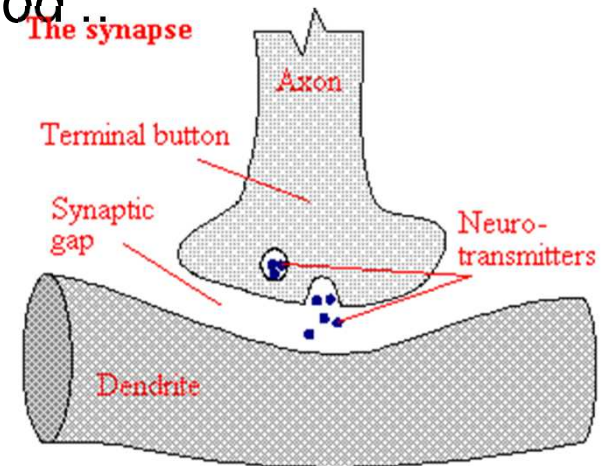
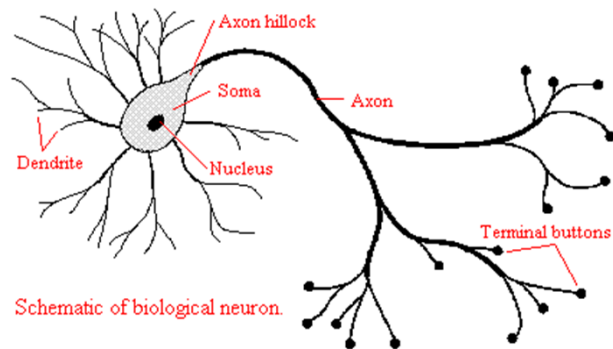
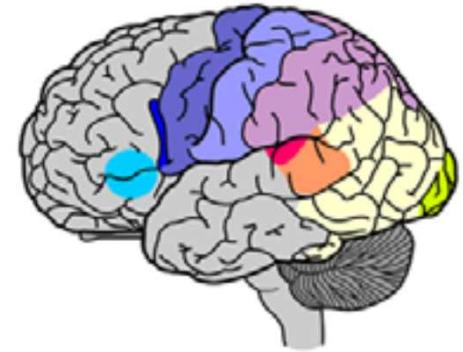


UNIVERSITY OF
LIVERPOOL



Some facts about the brain as a PC...

- The brain has ~100 billion neurons (10^{11}) – about 30 μ m large
 - Neuron Fan-in ~ $10^3 - 10^4$ (logic gates 2-4!)
 - complex dynamics - includes several time constants,
 - maintains a more complex internal state
 - output is a time-series of action potentials or 'spikes' - no information in amplitude!
- Massively parallel in nature
 - Typical 10^{15} interconnections
 - Total computation rate of about 10^{16} complex operations /sec (cf 10 P-FLOPs)
- Millisecond time frame of 'events'
- Low level function: 'reasonably well understood'
- High level function.....???????



Some other brains

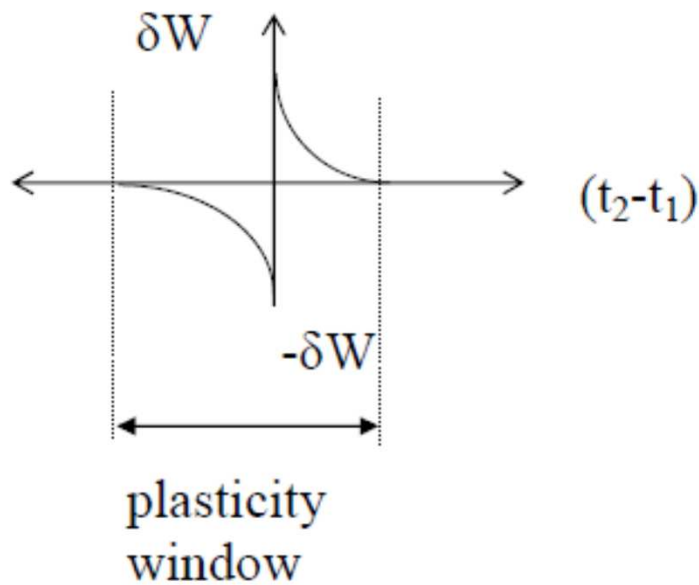
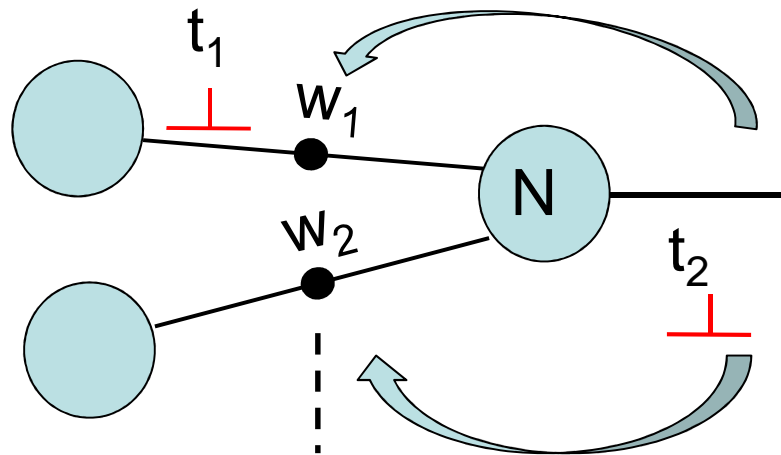
- A fly (1 grain of sugar a day to feed it!): 250 k neurons
- Honeybee (fantastic navigator!): 1 million neurons
- Rat (pretty smart animal): 55million neurons

- But how do the following work:
 - the arithmetic
 - Fault-tolerance
 - The parallelism (beat Moore's Law hands down)

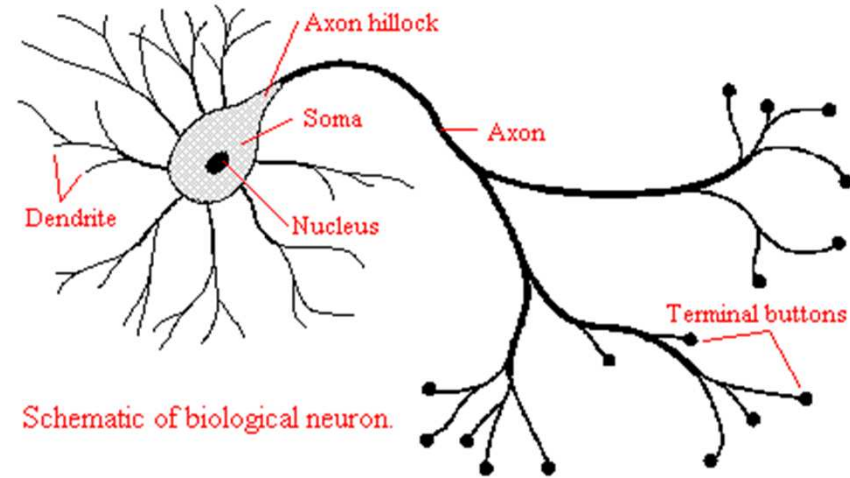
This is the inspiration!
But must find a simpler, scaleable, low power approach

Synapses and neurons

Spike-timing dependent plasticity



STDP learning rule



If spike, t_1 causes neuron, N to fire ($t_2 - t_1$ small)..

Weight W_1 may be increased

and W_2 etc decreased

Motivation


Create building blocks that can emulate biological functionality

Implement in mixed signal CMOS (cheap!)

Assess layout / scalability / systems functionality

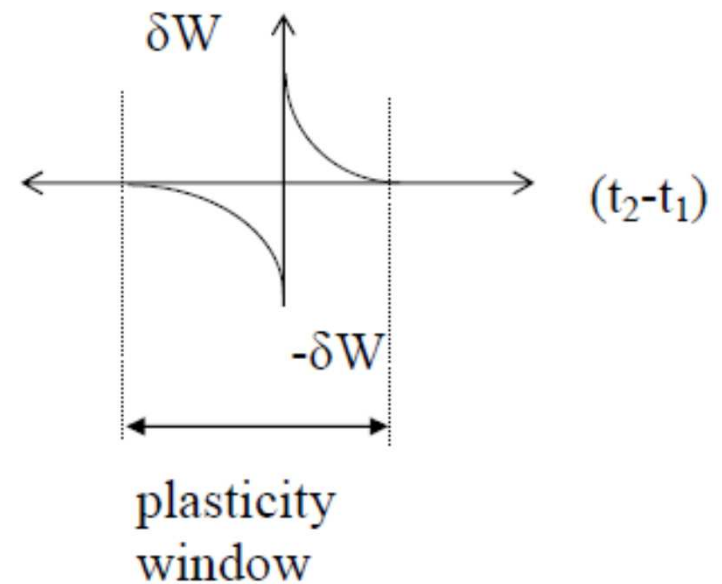
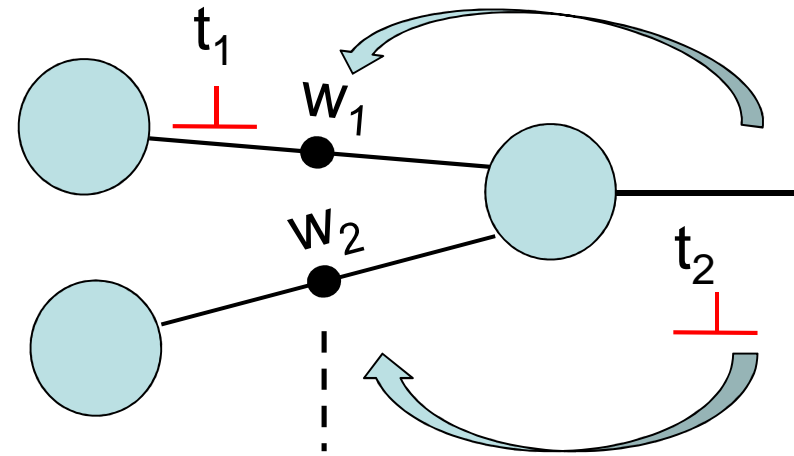
Circuits that can learn!

- Plasticity / decision circuits (STDP) / FG weight storage

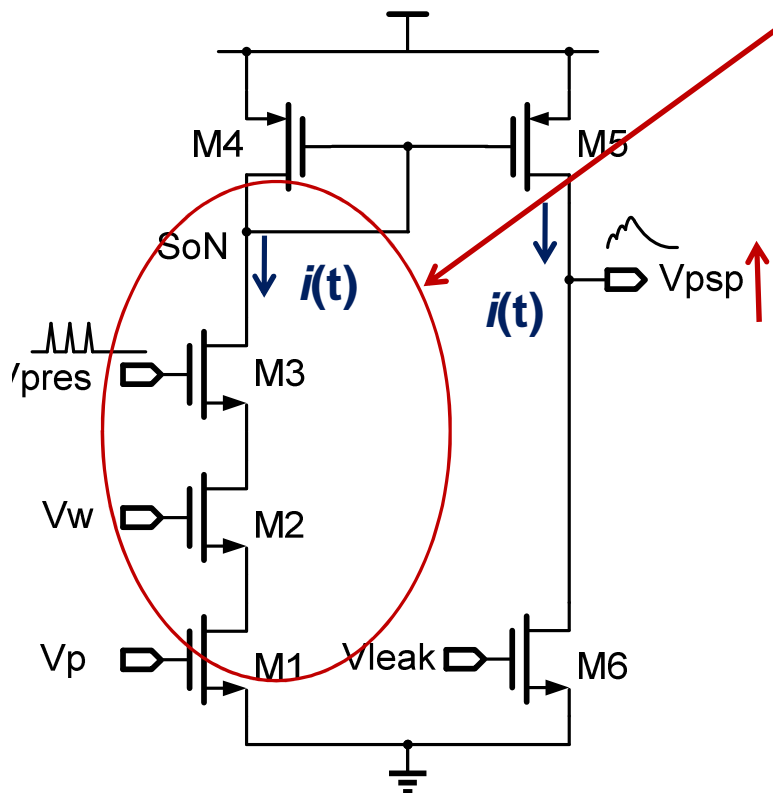
 Build large, useful electronic systems
learn more about 'brain computation'

Circuit Challenges

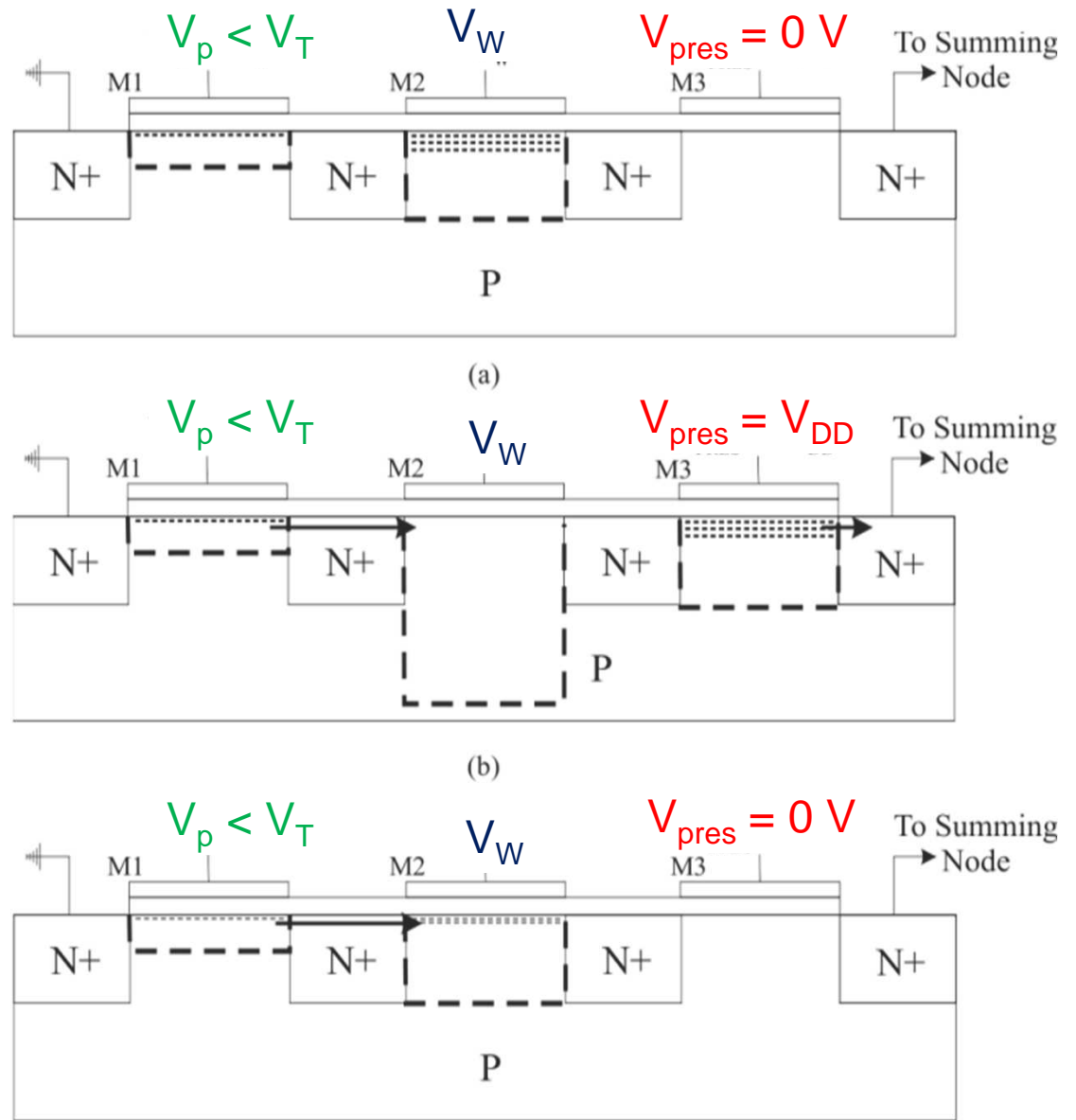
- Store and update weights
- Detect timing ($t_2 - t_1$)
- Axonal delay
- Low power operation
- Scale to VLSI
- Learn!



Dynamic synapse

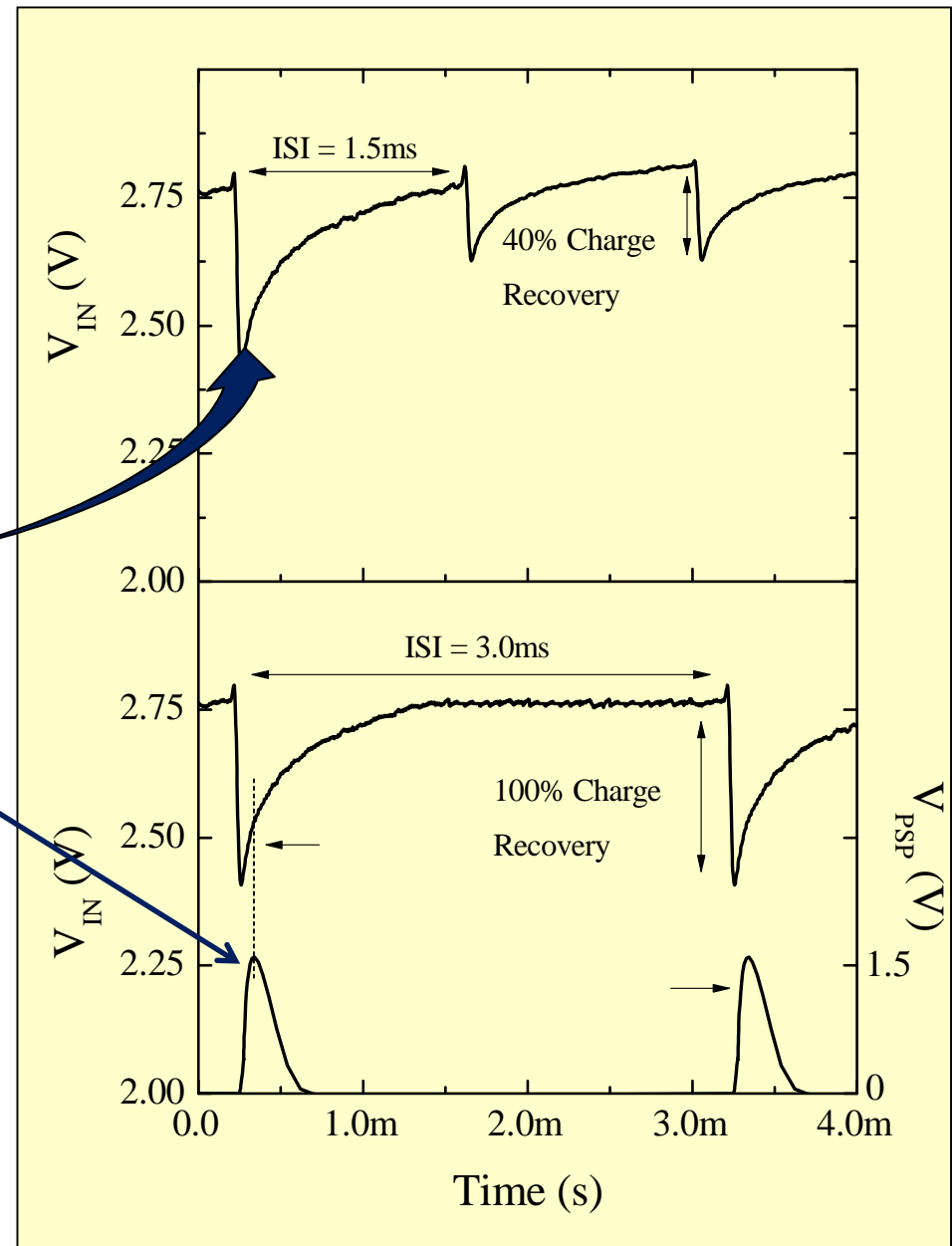
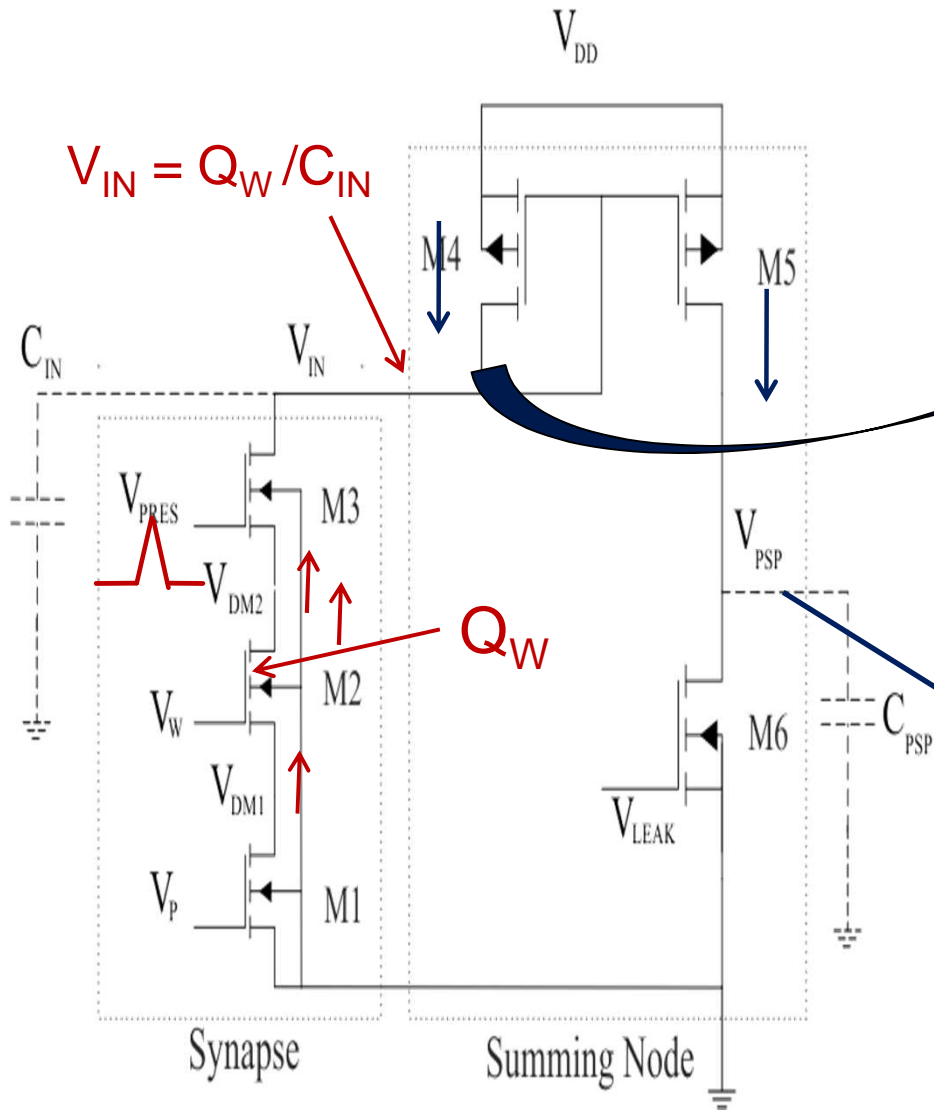


$V_{pres} \rightarrow$ on
 Charge sharing
 S of M2 increase
 \rightarrow M2 clamped 'off'
 Transient i , mirrored in M5

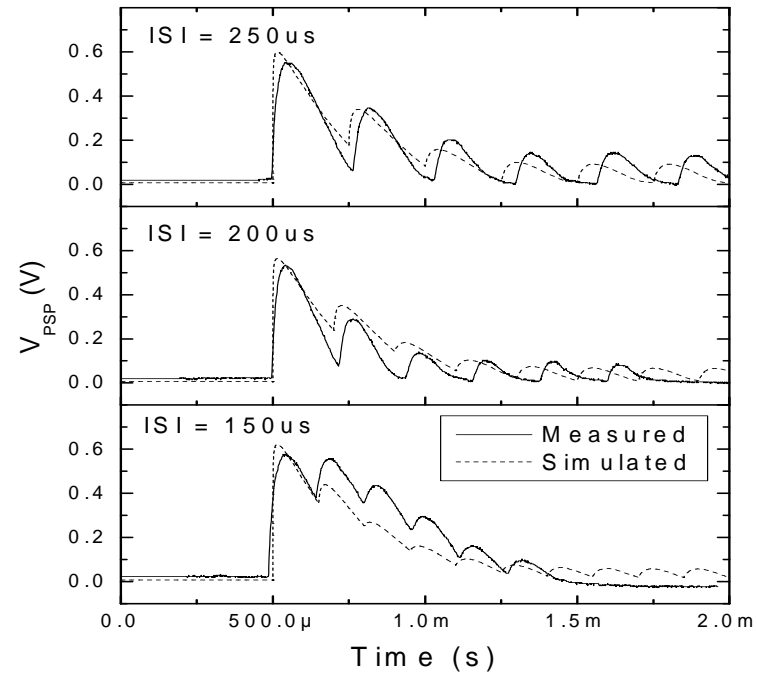
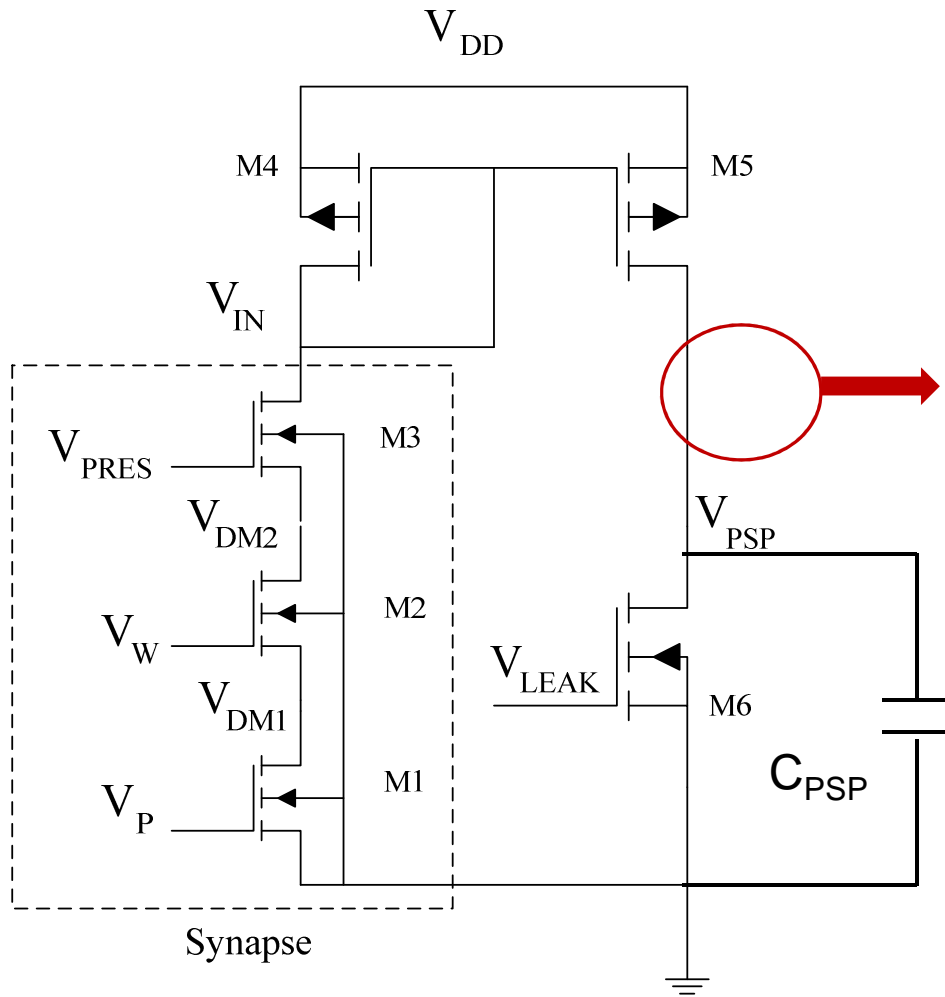


Dowrick et al.' IEEE Trans. On Neural Networks
 and Learning Systems, 23(10), p.1513 (2012)

How it works



Post-synaptic potentials

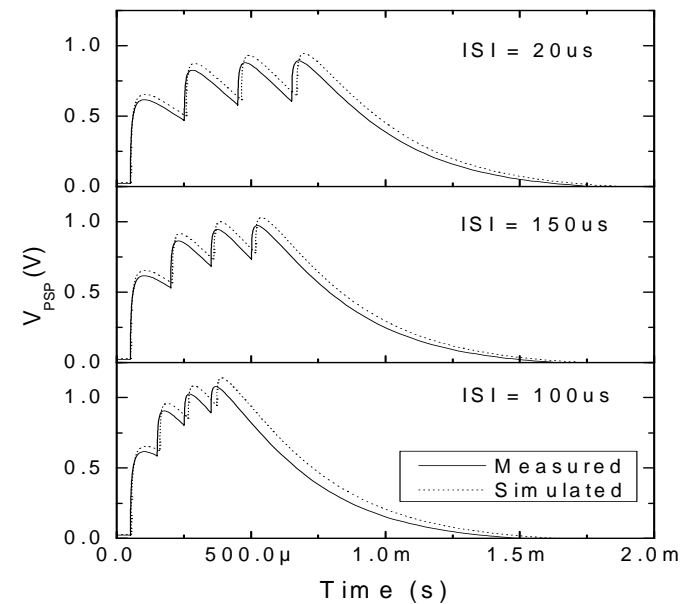


ISI

250us

200us

150us



20us

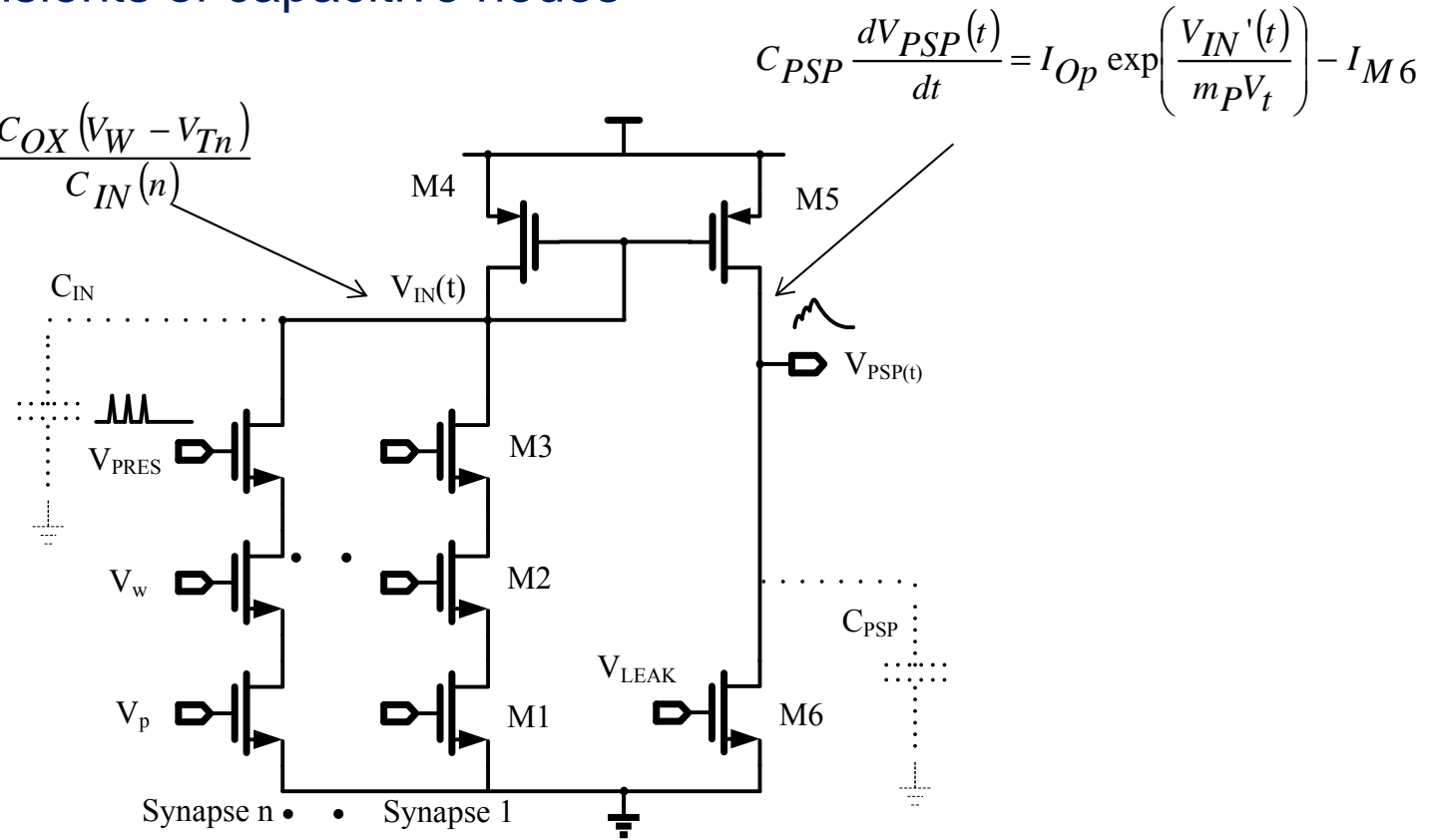
150us

100us

Fan-in: theory

Consider transients of capacitive nodes

$$\Delta V_{IN}(t) = \frac{Q_W}{C_{IN}(n)} = \frac{C_{OX}(V_W - V_{Tn})}{C_{IN}(n)}$$



Rise time

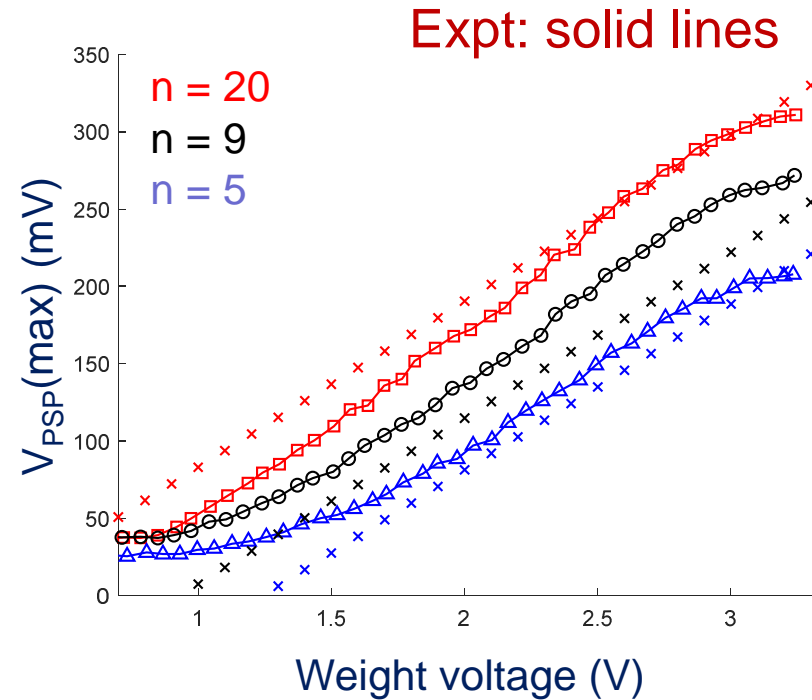
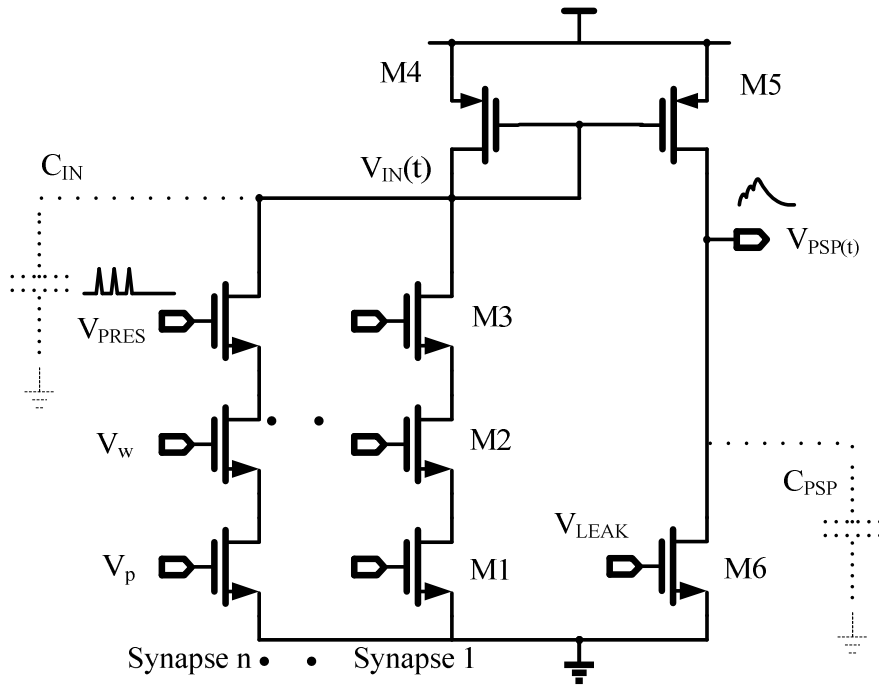
$$\tau_R = \left[\frac{I_{Op}}{m_p V_t C_{IN}} \right]^{-1} \left\{ \frac{I_{On}}{I_{Op}} \exp\left(-\frac{V_{LEAK}}{m_n V_t}\right) - \exp\left(-\frac{V_{IN}'(0)}{m_p V_t}\right) \right\}$$

Post-synaptic potential

$$V_{PSP}(t) = m_p V_t \frac{C_{IN}}{C_{PSP}} \ln \left[1 + \frac{I_{Op}}{m_p V_t C_{IN}} \exp\left(\frac{V_{IN}'(0)}{m_p V_t}\right) t \right] - \frac{I_{M6}}{C_{PSP}} t$$

Fan-in

$$V_{PSPMAX} = m_p V_t \frac{C_{IN}(n)}{C_{PSP}} \ln \left[1 + \frac{I_{Op}}{m_p V_t C_{IN}(n)} \exp \left(\frac{V_{IN}'(0)}{m_p V_t} \right) \tau_R \right] - \frac{I_{M6}}{C_{PSP}} \tau_R$$

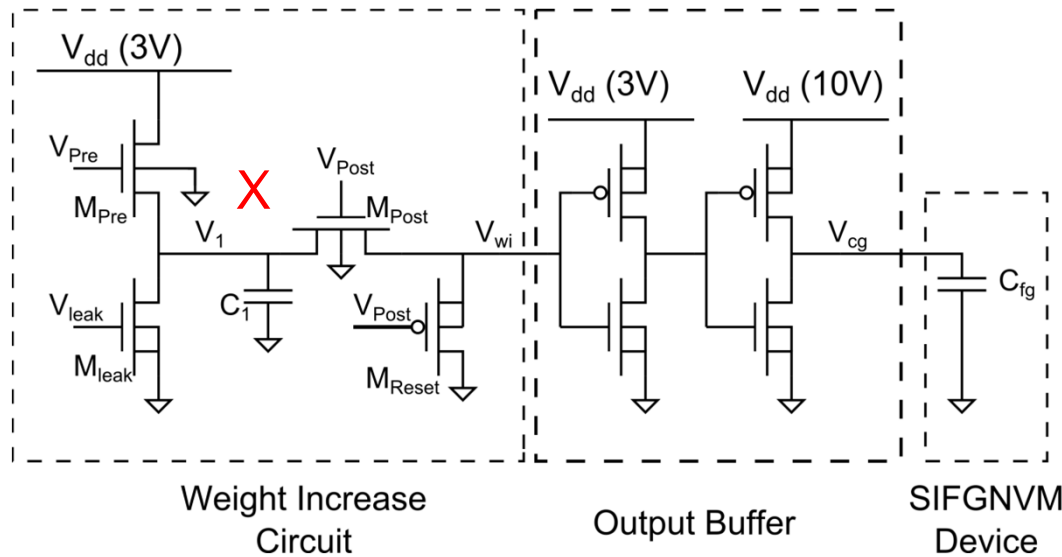


Conclusion: Fan-in intrinsic limit > 10⁵ !

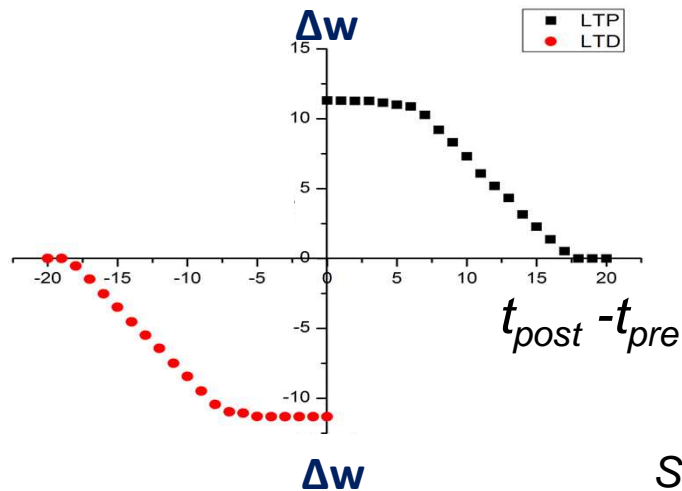
Practical limit is set by layout / interconnect

Compact decision circuits (STDP)

Weight Increase, WI, Circuit Block, Output Buffers and SIFGNVM Device



Similar circuit for weight decrease



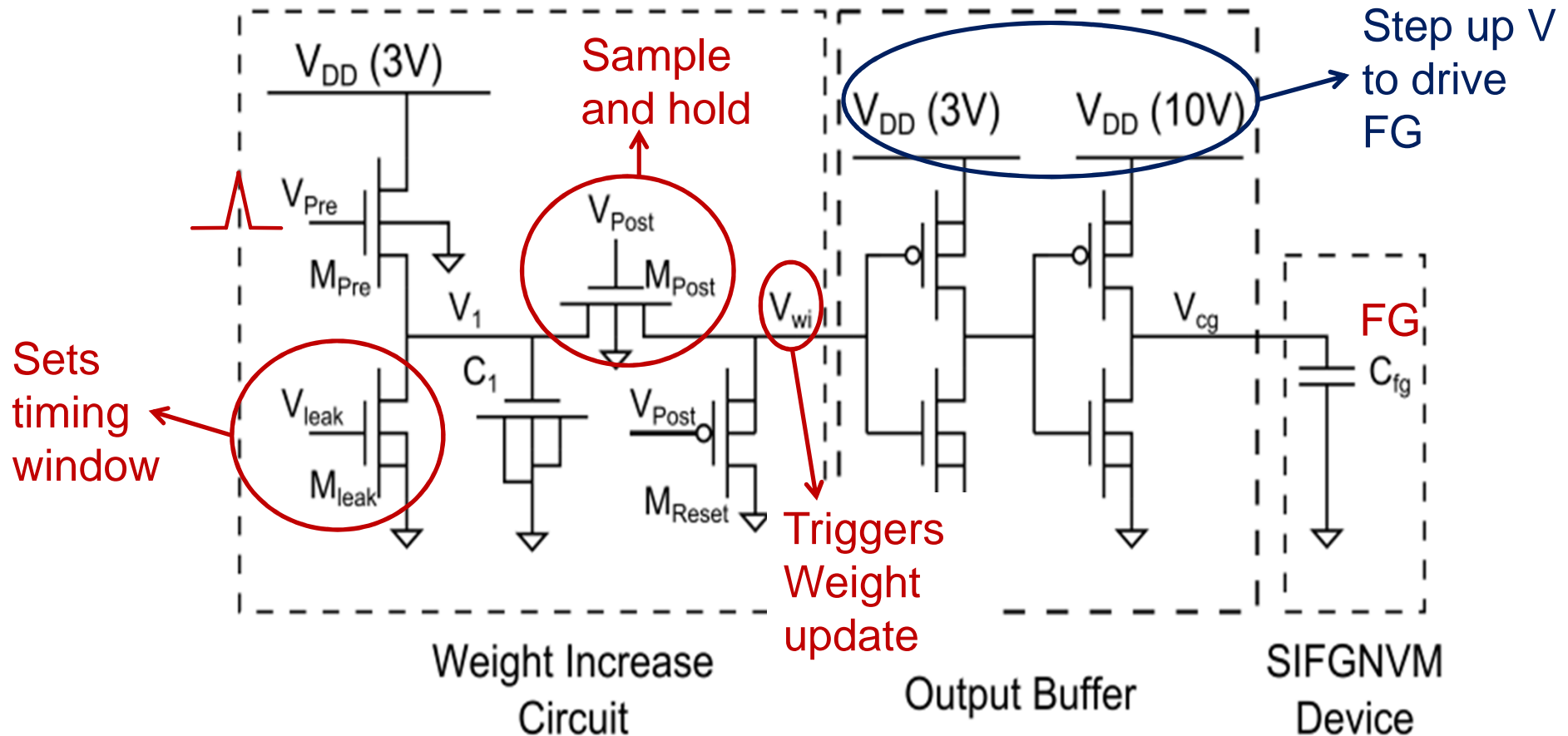
Pass transistors gated by V_{pre} , V_{post} charge node X

Decay via sub-Vth MOST

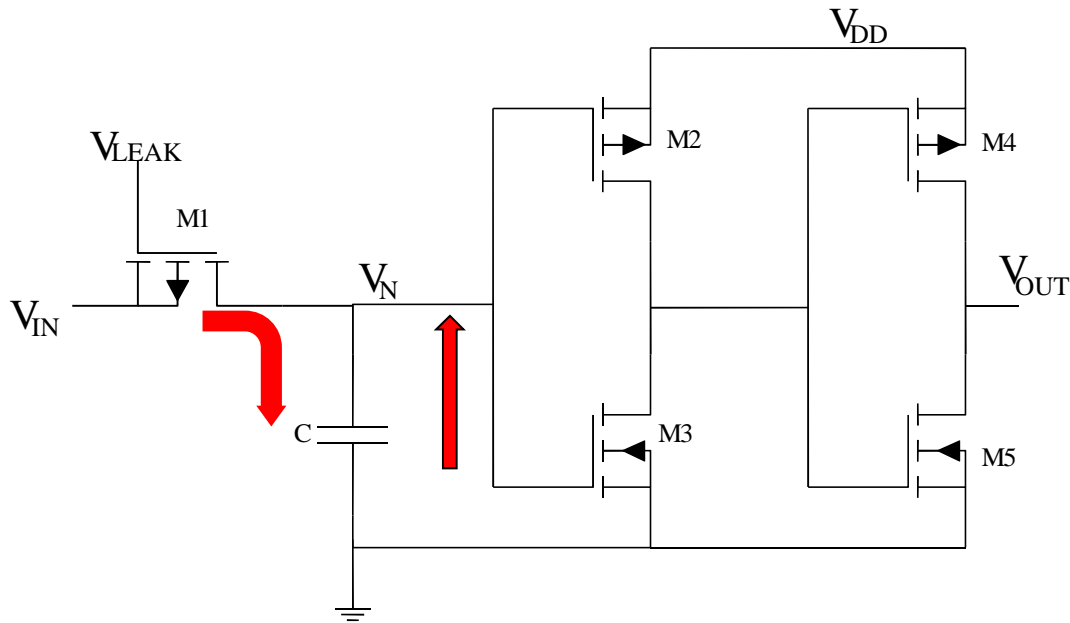
Sets plasticity 'window'

How it works: WI Block Operation Pre-Post Spiking Event

- When a presynaptic spike occurs (V_{Pre})
 - V_1 is pulled up to $3V - V_{TMpre}(V_1)$, C_1 charges via M_{pre}
 - C_1 Slowly discharges via sub-threshold M_{leak}
 - V_{post} triggers the sample/hold as some time, t after V_{pre}

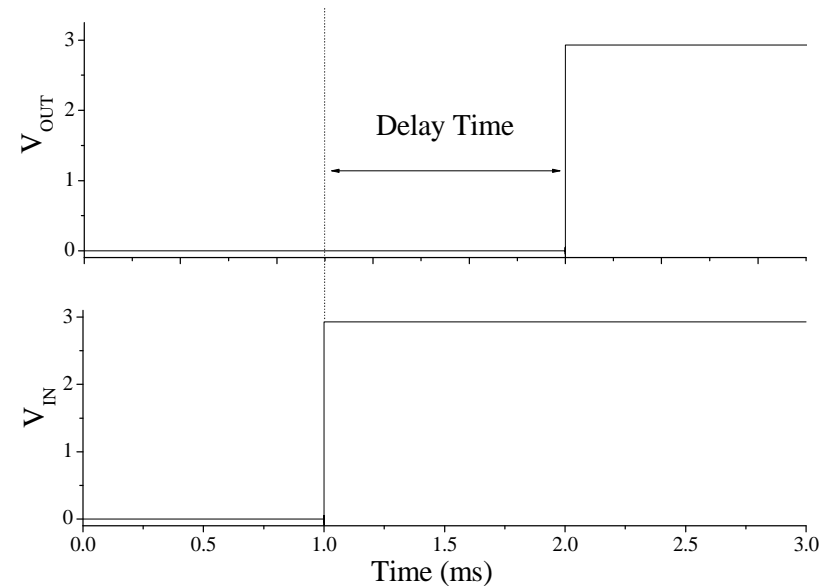


Axonal delay



- M_1 operates subthreshold
- Slow charging of C
- V_N rises and inverters turn on
- Tune delay with V_{LEAK}

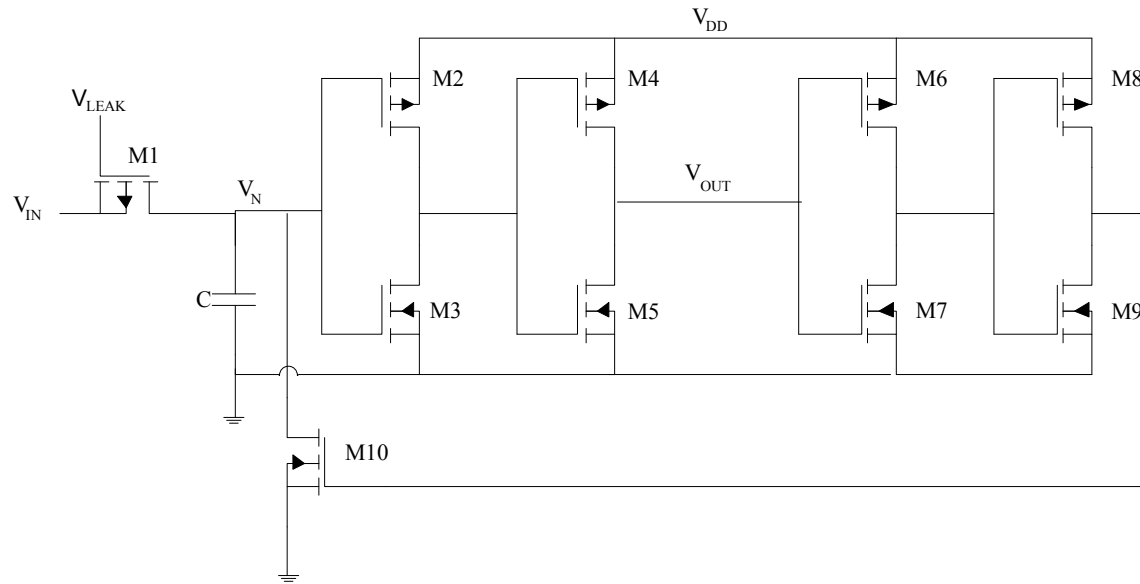
$$t_d = \frac{C}{I_0 \exp\left(\frac{V_{GSM1}}{mV_{th}}\right)} (V_{TRIG} - V_{N0})$$



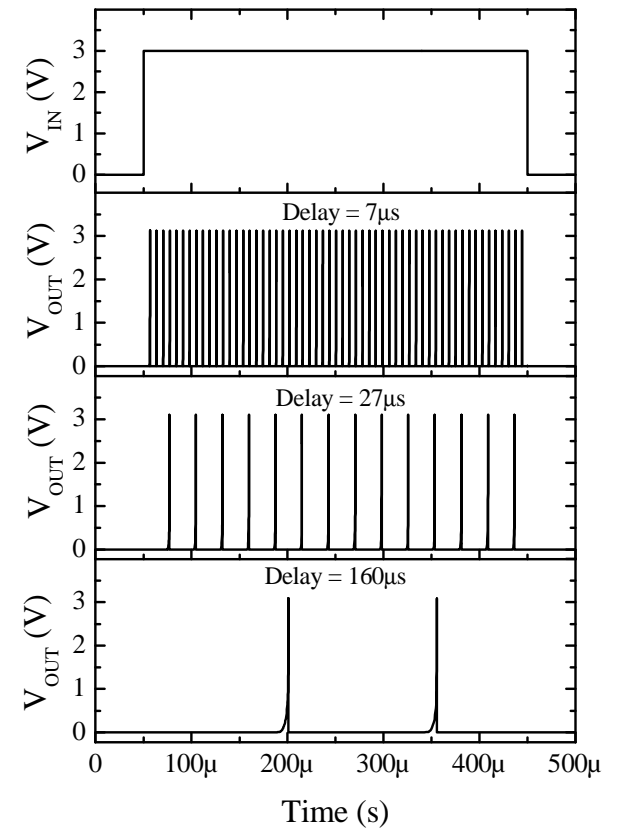
Dowrick et al, Neurocomputing, 2012

<http://dx.doi.org/10.1016/j.neucom.2012.12.004>,

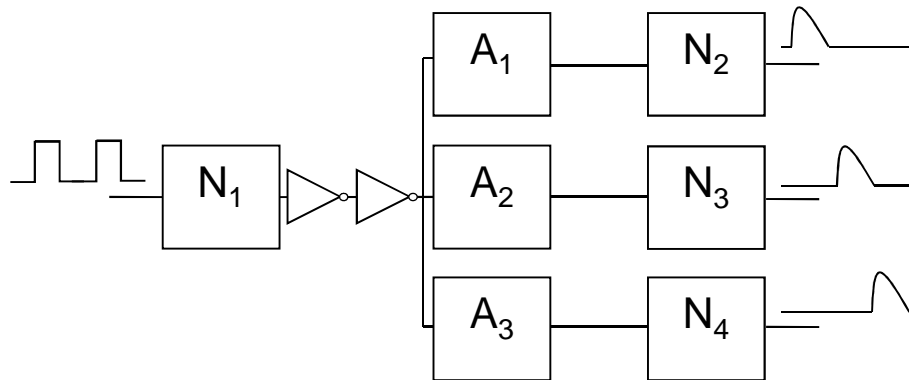
Pulse burst creation



Add feedback (M_{10})
Define pulse trains



Integrate axon delays (A) into paths

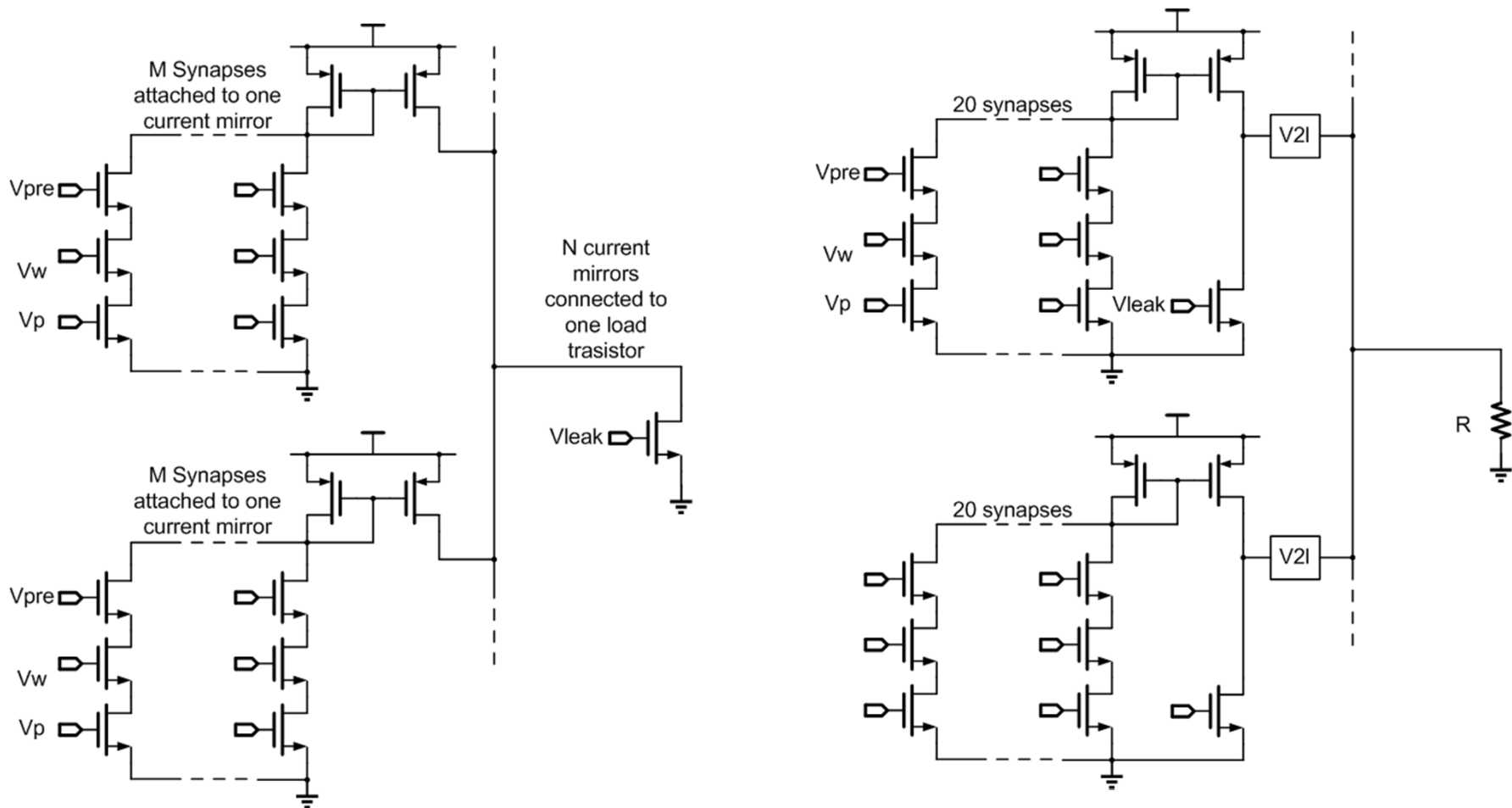


Dowrick et al, Neurocomputing, 2012

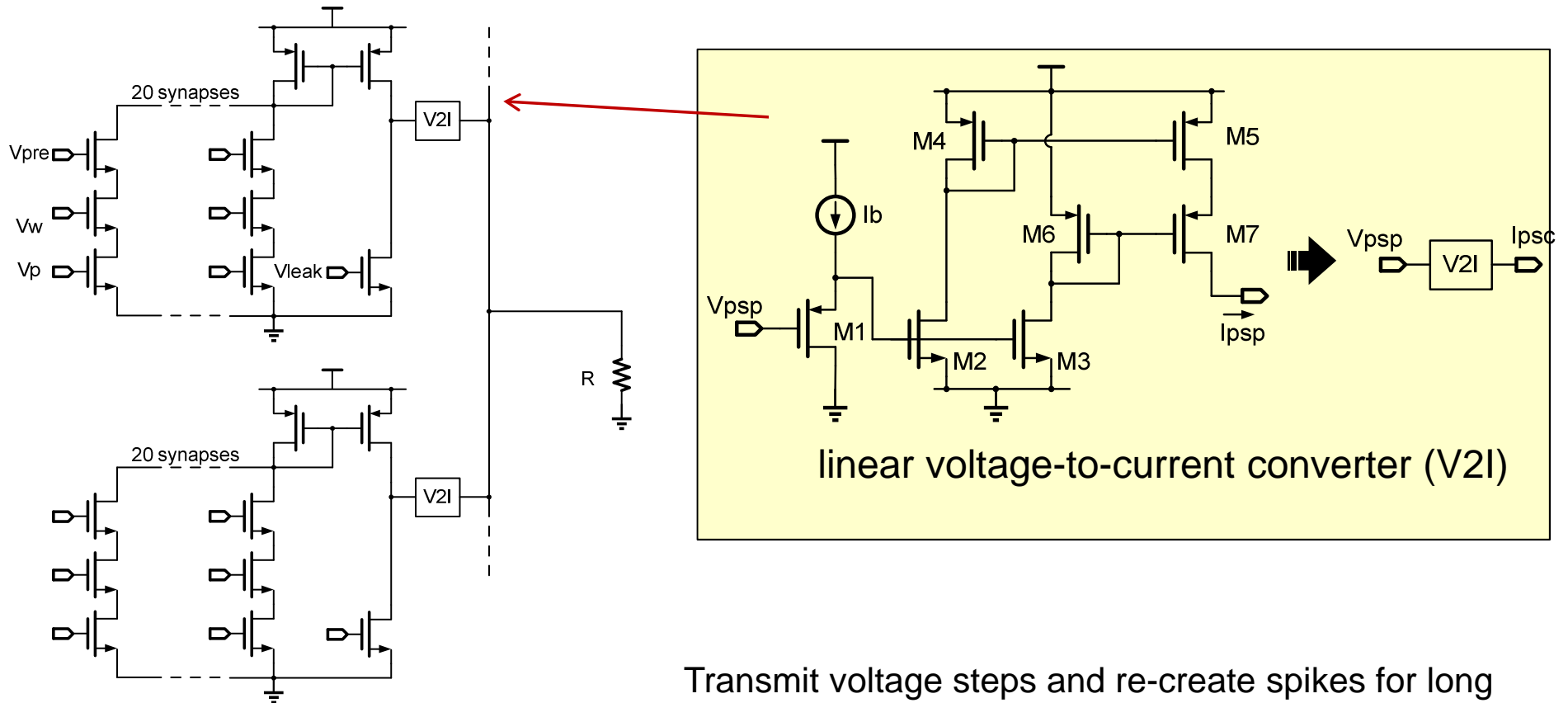
<http://dx.doi.org/10.1016/j.neucom.2012.12.004>,

Scaling

- Two solutions: sum voltages or sum currents

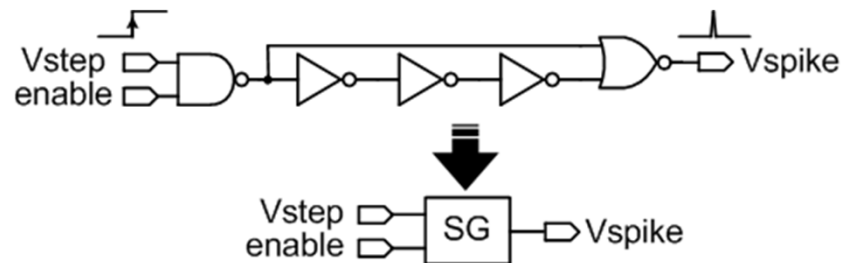


Scaleability: easier to sum currents



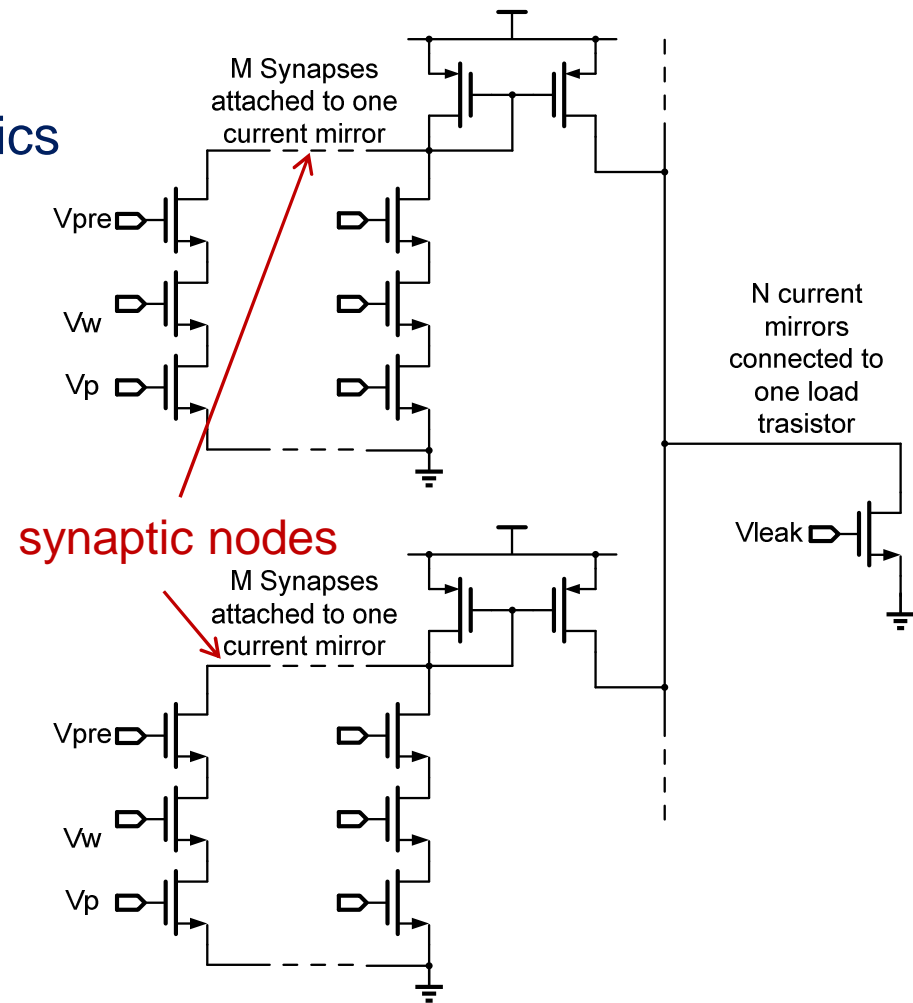
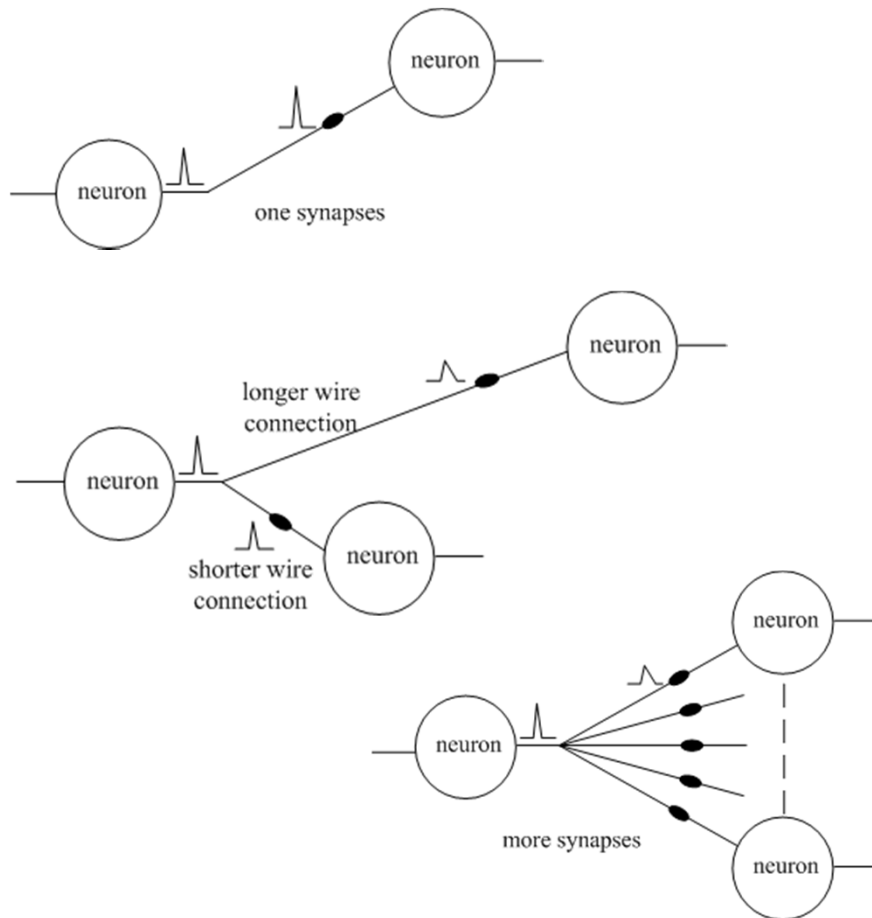
But added complexity!

Transmit voltage steps and re-create spikes for long interconnect



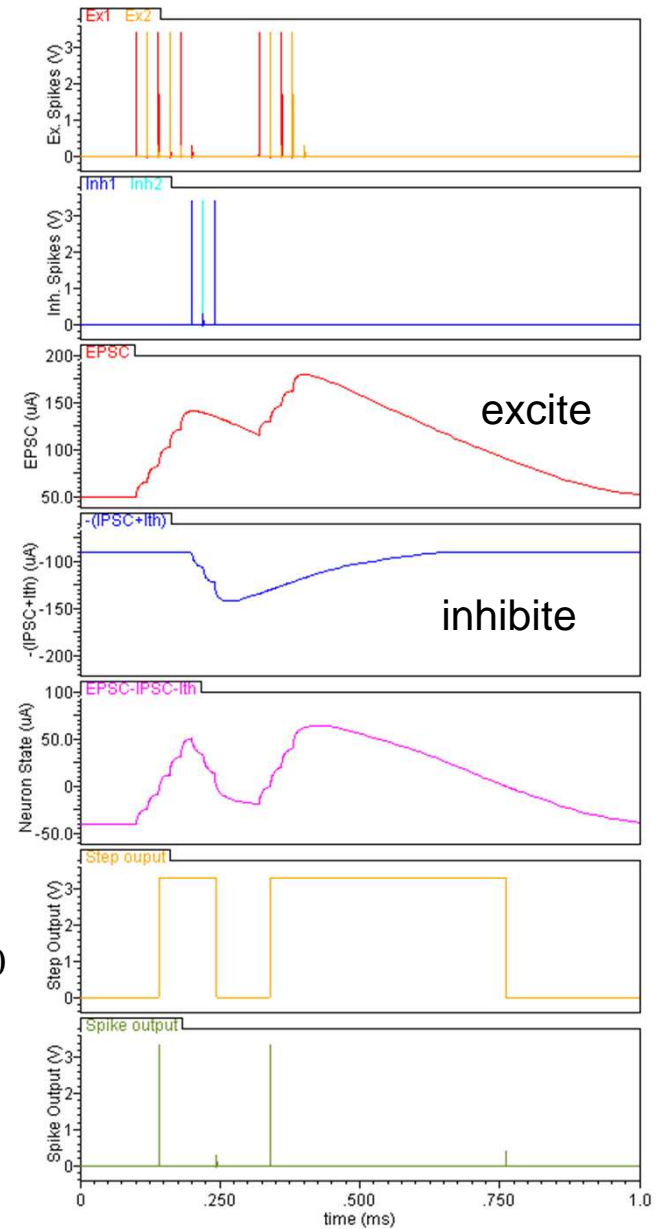
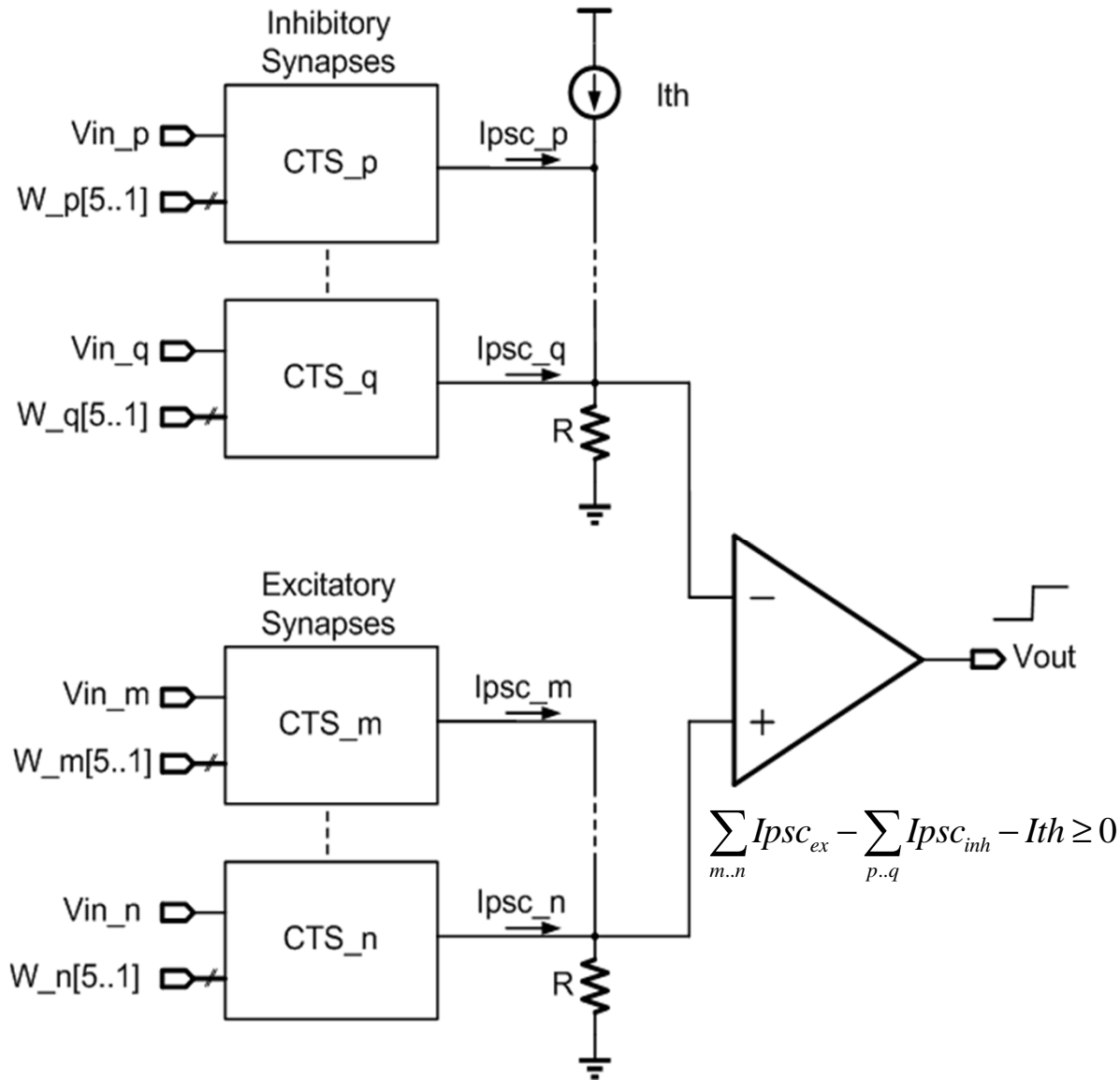
Scaling: circuit issues

Large synapse fan-out problem:
non uniform spike inputs due to parasitics
non-linearities occur in currents



Hope it all comes out in the wash!
Nature is messy as well

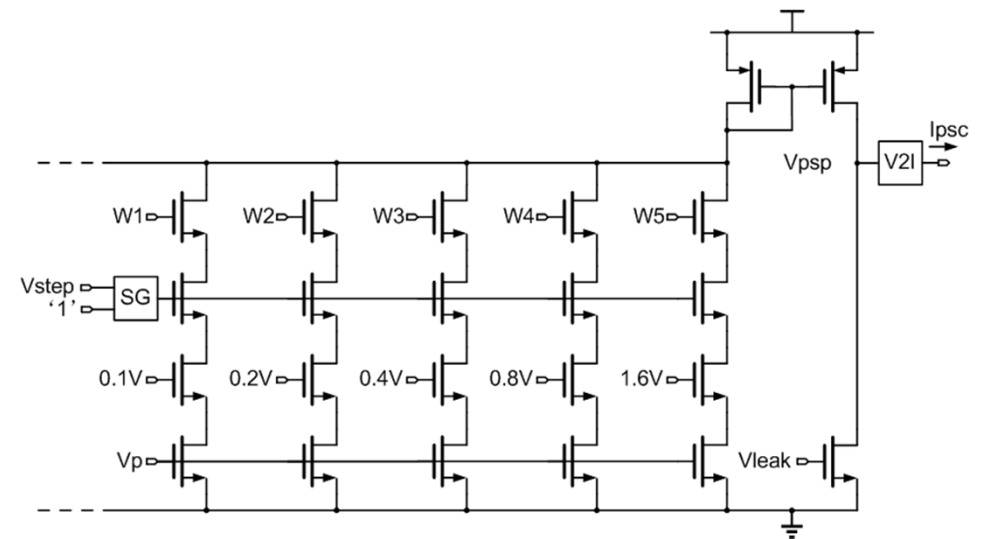
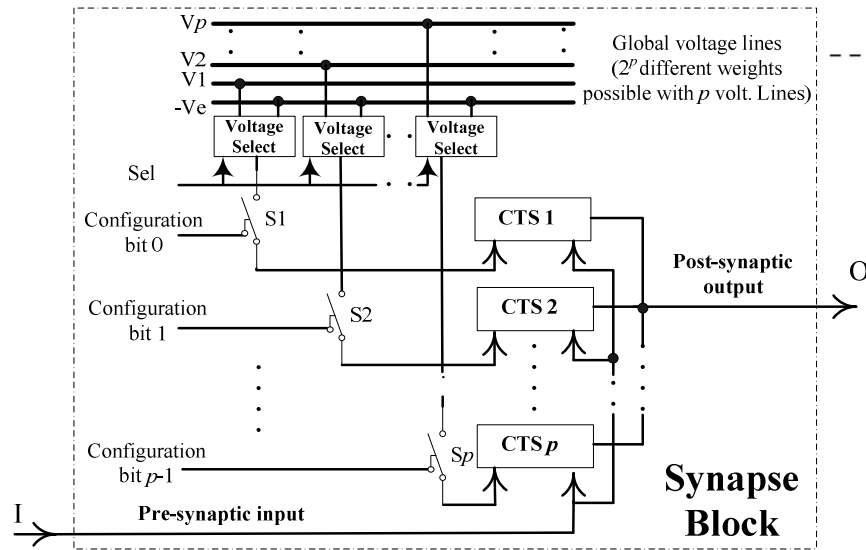
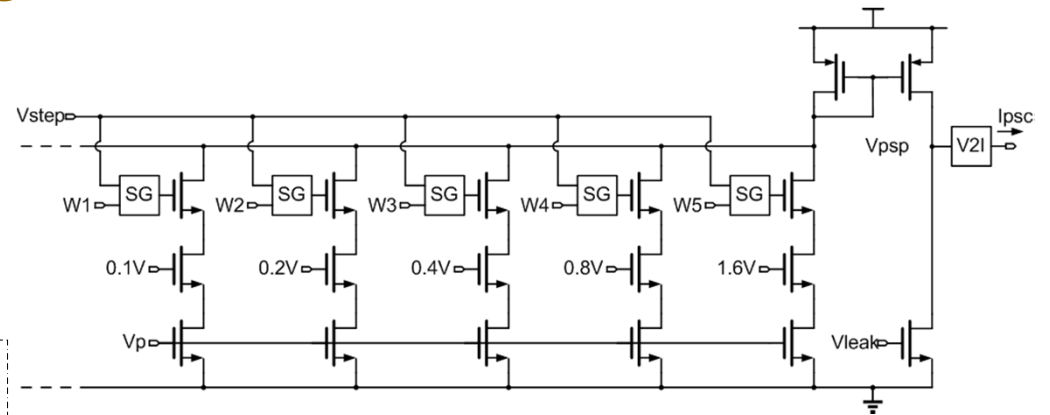
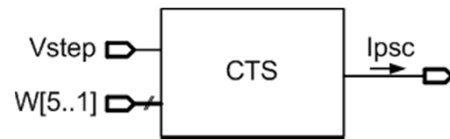
Neurons with excitatory and inhibitory synapses



Programmable weights

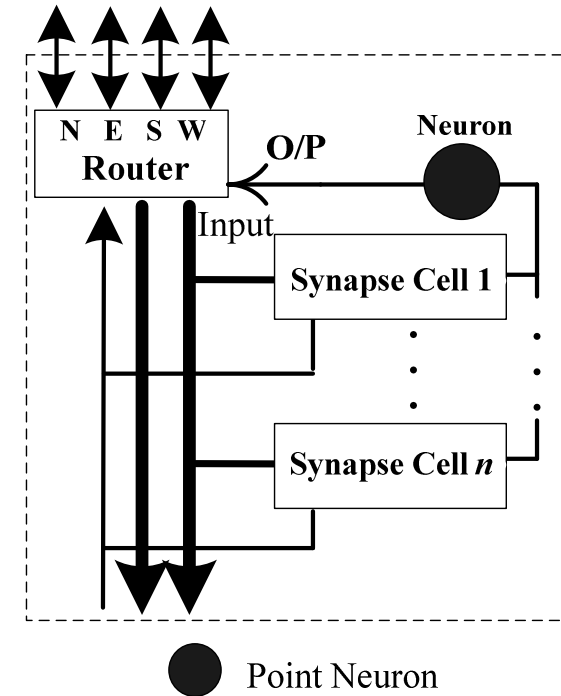
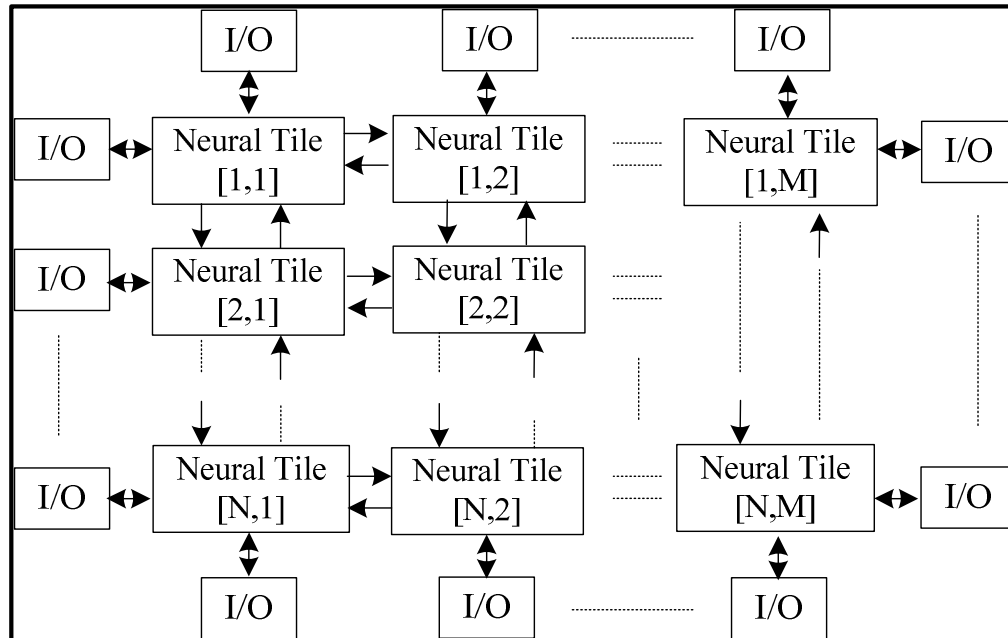
- Analog weight
 - Good: Continuous weight value, compact analog storage circuit
 - Bad: Inaccurate, require bias reference circuit and complex control circuit for high resolution, also require high voltage rail and undocumented
 - technology feature
- Digital weight
 - Good: accurate, mature digital memory technology, easy to program
 - Bad: discrete quantitative weight, require more space

Programmable weight



Embrace: an alternative approach

- Network-on-chip address the issues of scalability and connectivity between components.
- Low-area/power spiking neuron cells with associated training provides neural computing capability.



- 2-dimensional array of interconnected neural tiles + I/O blocks.
- Neural tiles connected in North, East, South and West.
- Tile can be programmed to realise neuron-level functions.

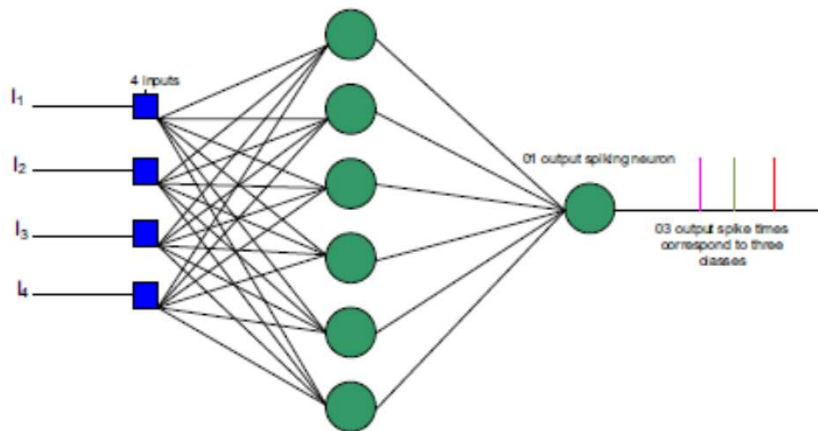
Harkin et al, Int. Jnl of Reconfigurable Computing, doi:10.1155/2009/908740 (2009)

Slide courtesy of Jim Harkin

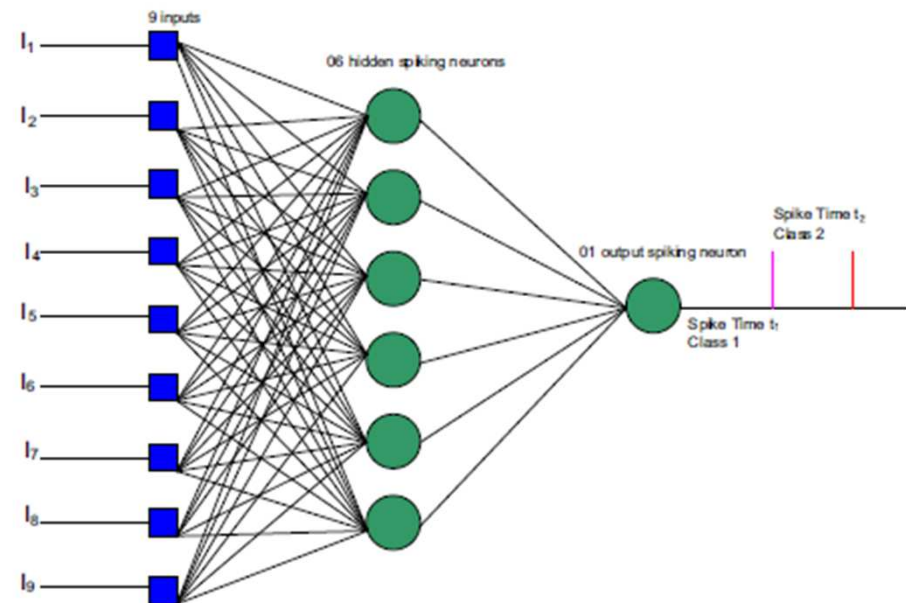
Evaluation

- Learning in software (calculate weight values)
 - Fit the experimental synapse results
- Solve benchmark problems
 - Wisconsin breast cancer (WBC) dataset
 - IRIS dataset
- Temporally encoded input values

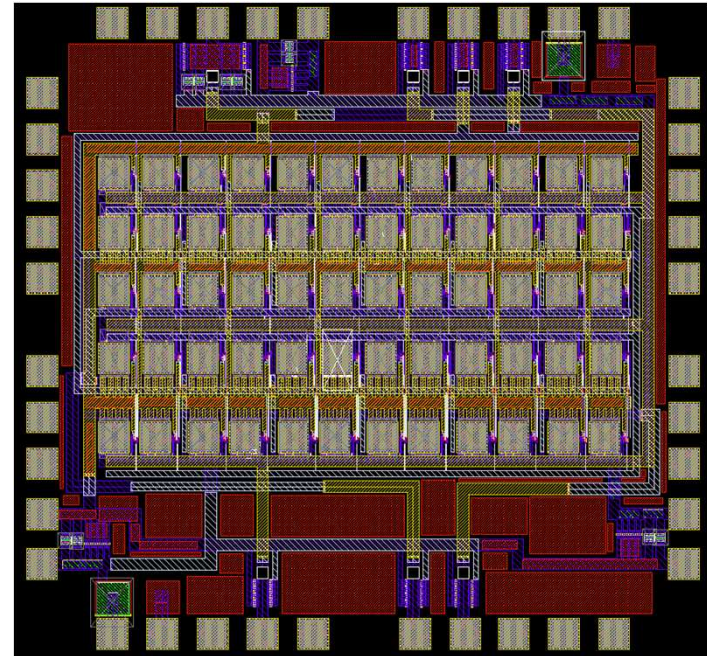
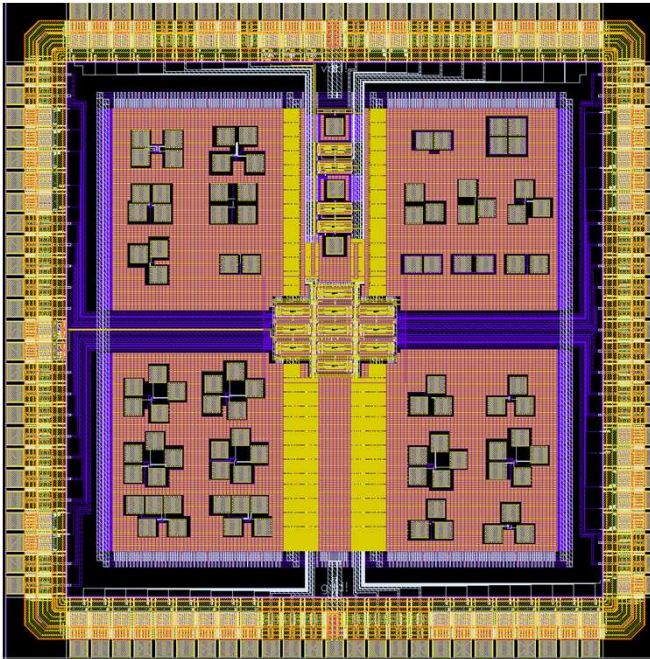
SNN architecture: IRIS dataset



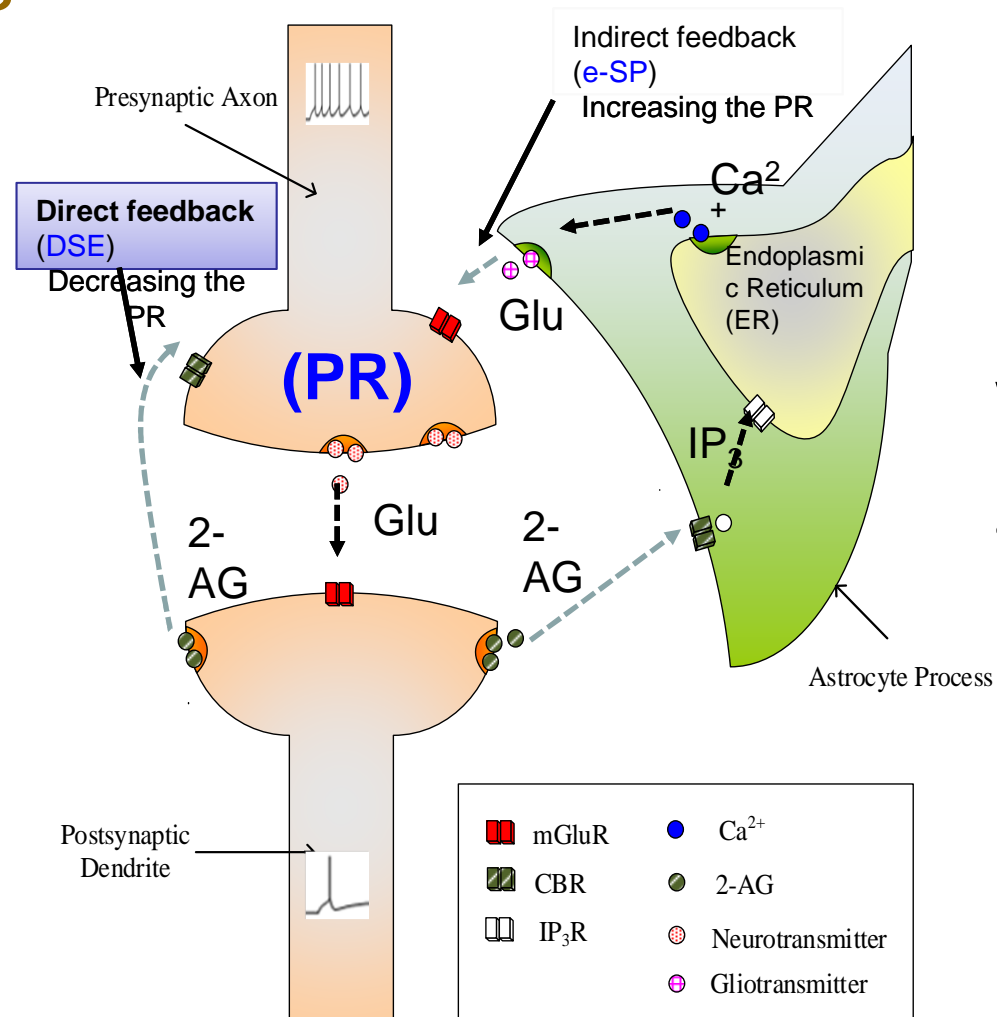
SNN architecture: WBC dataset



Circuits fabricated in AMS 0.35, mixed signal CMOS



Astrocytes

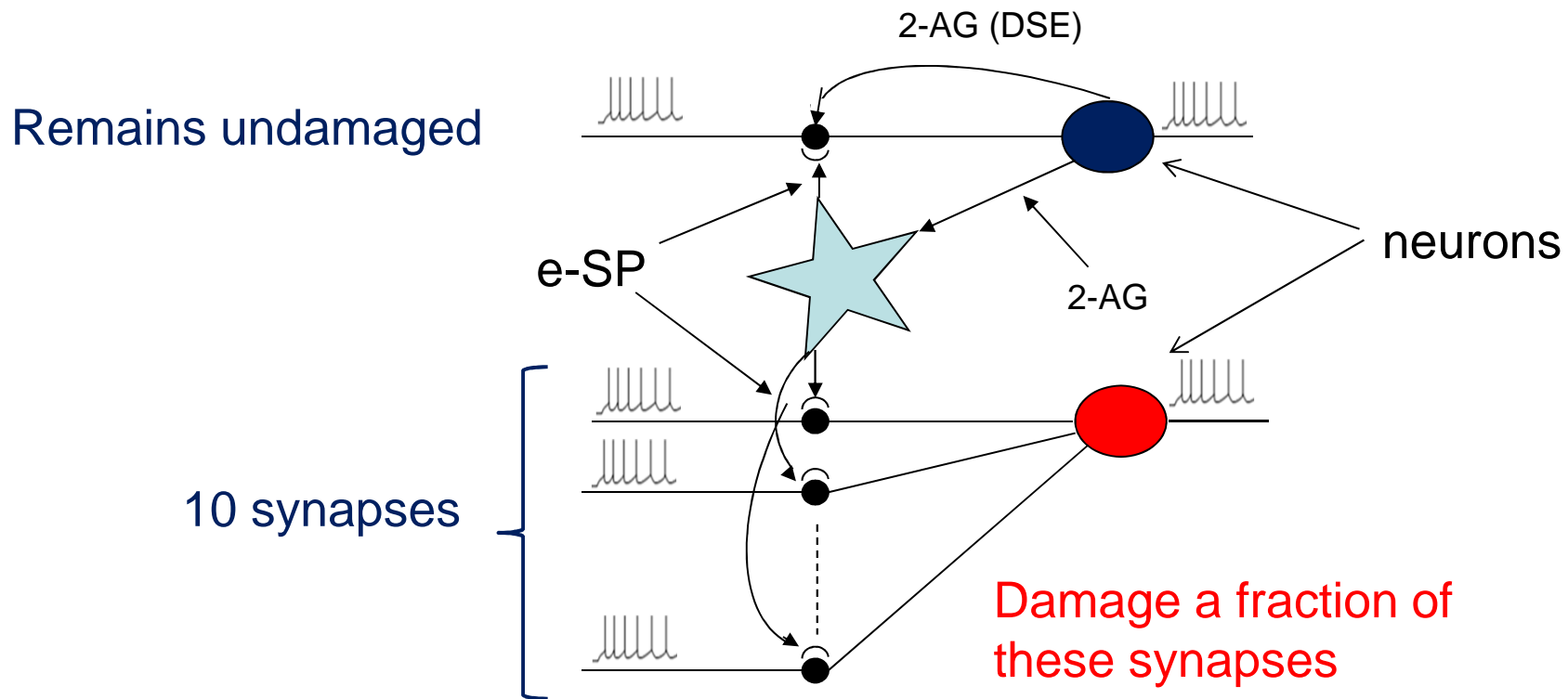


Study transport within astrocyte process and between neuron/astrocyte

Breslin et al, PLoS Computational Biology, doi.org/10.1371/journal.pcbi.1006151, May (2018)

Slide courtesy of Professor Liam McDaid

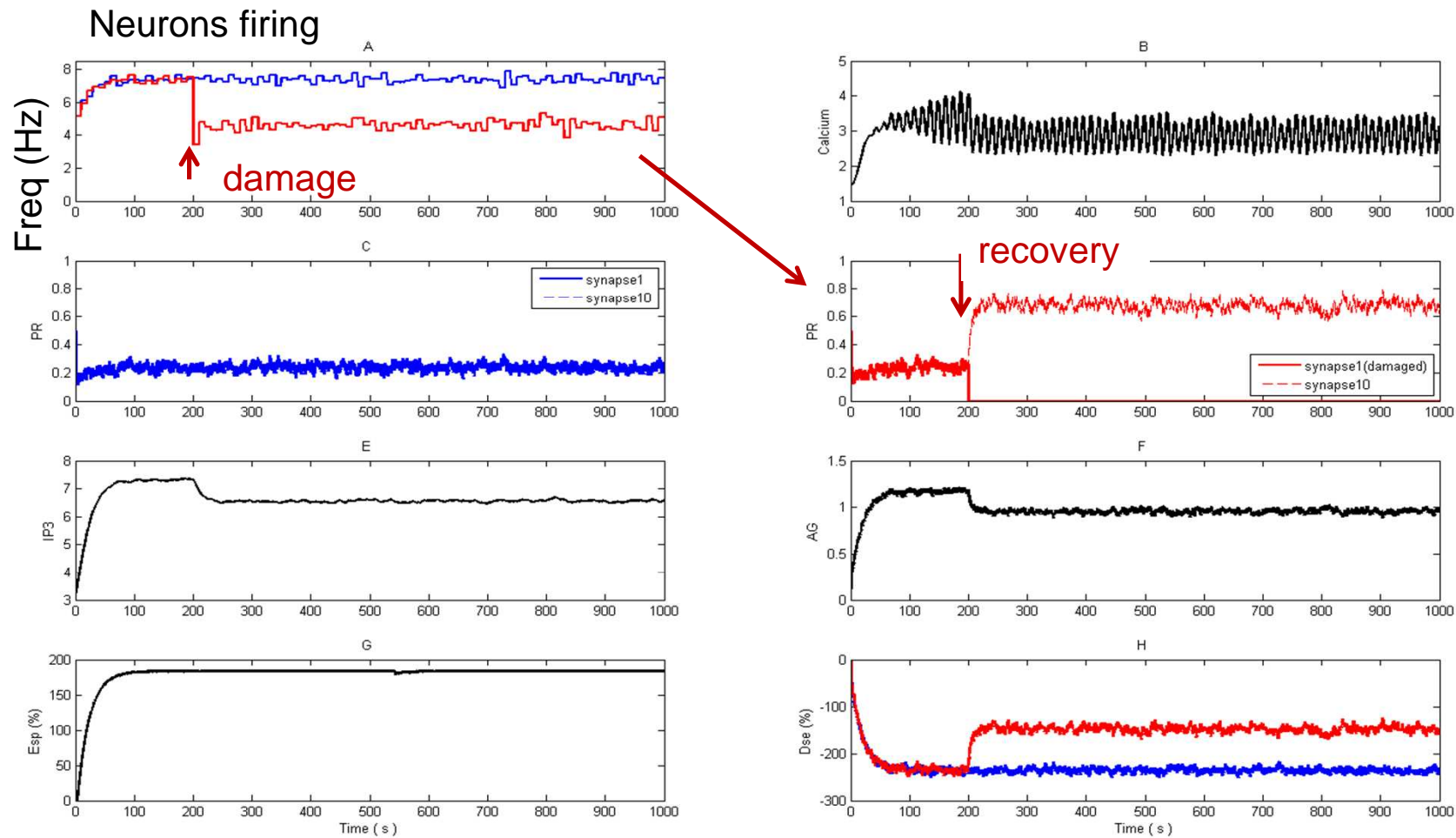
Endocannabinoid Mediated Self-Repair



Wade, McDaid et al, *Frontiers in computational neuroscience*, v6, Art 76 (2012)

Slide courtesy of Professor Liam McDaid

Astrocytes mediate self-repair



- Astrocyte ‘forces’ undamaged neurons to ‘work harder’
- Opens up STDP window – restarts learning

Slide courtesy of Professor Liam McDaid

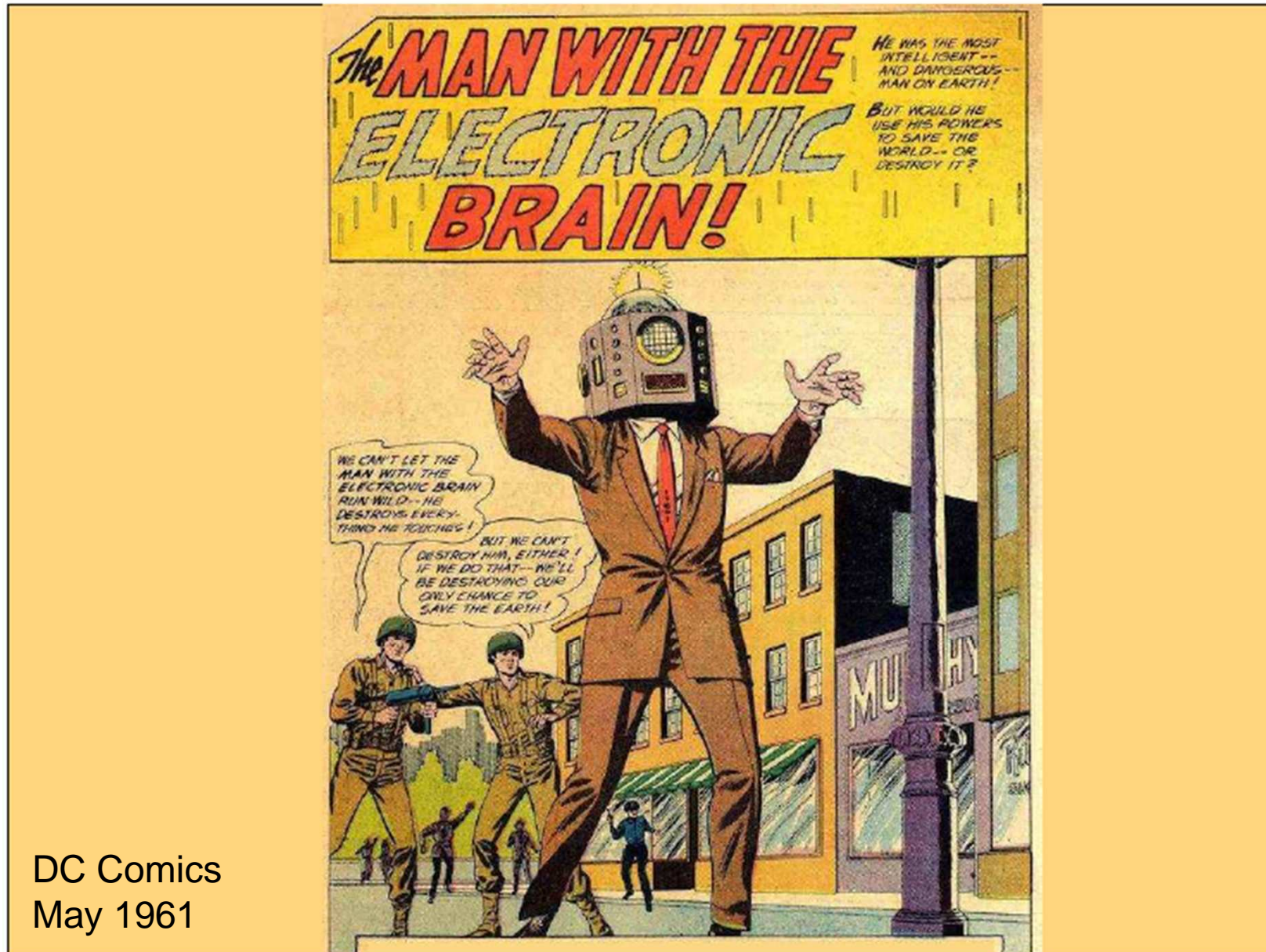
Wade, McDaid et al, *Frontiers in computational neuroscience*, v6, Article 76 (2012)

What we learnt..

- Can build compact analogue circuits that emulate aspects of biology with a degree of success (better than in software? – potentially much faster)
- Getting them to learn is another matter..
 - Need feedback
 - Weight update
 - Starts to get very complicated...
- A lot of redundancy once the circuit has ‘learnt’
- Scaling soon results in a huge amount of interconnect

Need software/hardware combination – learning in software

Still some way to go before....



Thanks to

LJ McDaid (*lj.mcdaid@ulster.ac.uk*)

J Harkin (*jg.harkin@ulster.ac.uk*)

T Dowrick (PhD)

A Smith (PhD)

S Chen (PhD)

S Zhang (post-doc)

A Ghani (post-doc)

Funding: EPSRC, NAP, EPSRC-DTA awards, Dorothy Hodgkin scholarship