Field Programmable Arrays for Neuromorphic Computation

Alister Hamilton

14th & 15th June 2010
UPC, Barcelona

Talks Schedule

14th June 2010: Neuromorphic Systems in Analogue VLSI: developments at the University of Edinburgh

15th June 2010: Programmable Analogue VLSI Architectures: two novel approaches

Programmable Analogue VLSI: Architectures: two novel approaches

Alister Hamilton

15th June 2010
UPC, Barcelona

Agenda

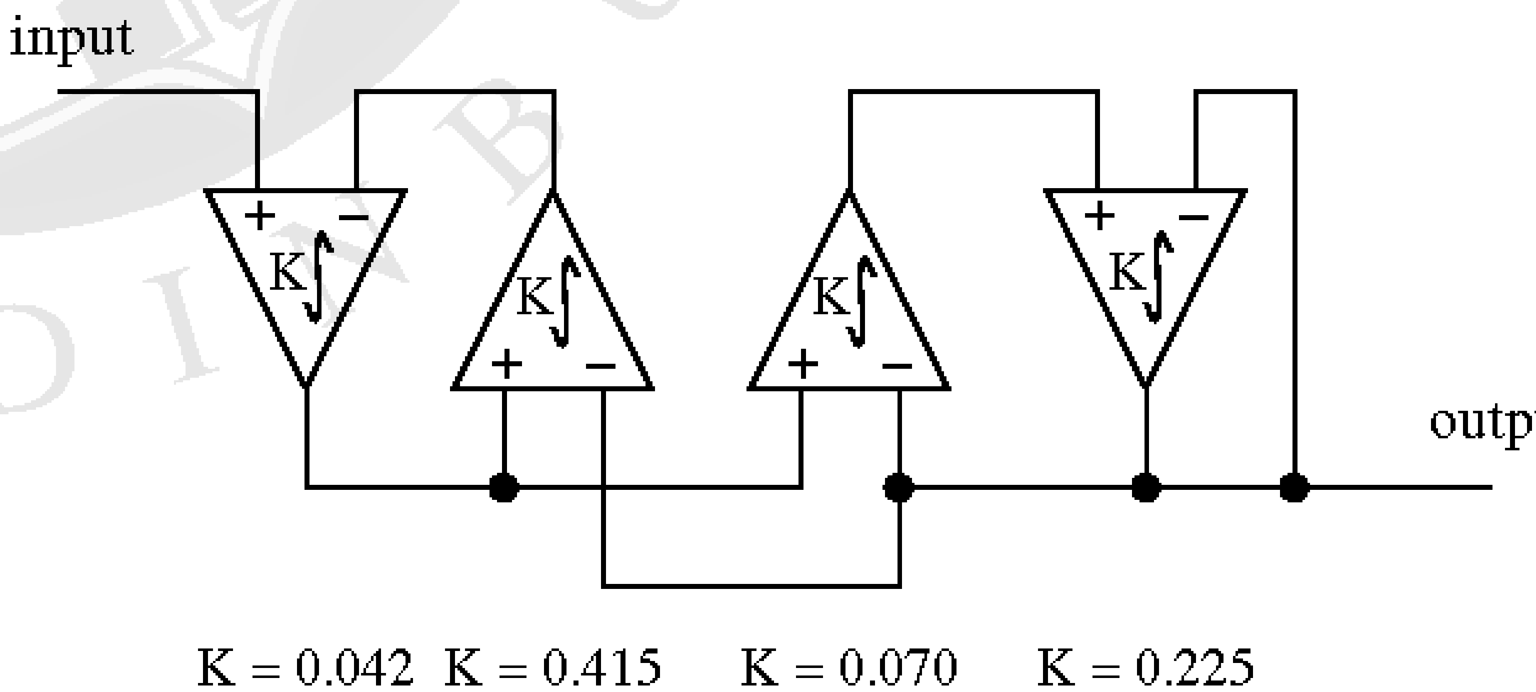
- Why programmable analogue VLSI?
  - Influence of previous research interests on a new research direction at the University of Edinburgh
- Early programmable analogue VLSI work
  - ***Palmo:*** pulse based programmable analogue VLSI.
- Current work at the University of Edinburgh
  - Programmable analogue VLSI architectures based upon ***event coding***.

Programmable analogue VLSI

- Inspiration for programmable analogue VLSI came from study of ***wavelets***.
- ***Wavelets*** considered as pre-processors for neural network analogue VLSI.
- ***Wavelets*** may be implemented using arrays of low pass filters.
- To implement ***wavelets***, need to implement low pass filters
  - useful if these filters can be made programmable

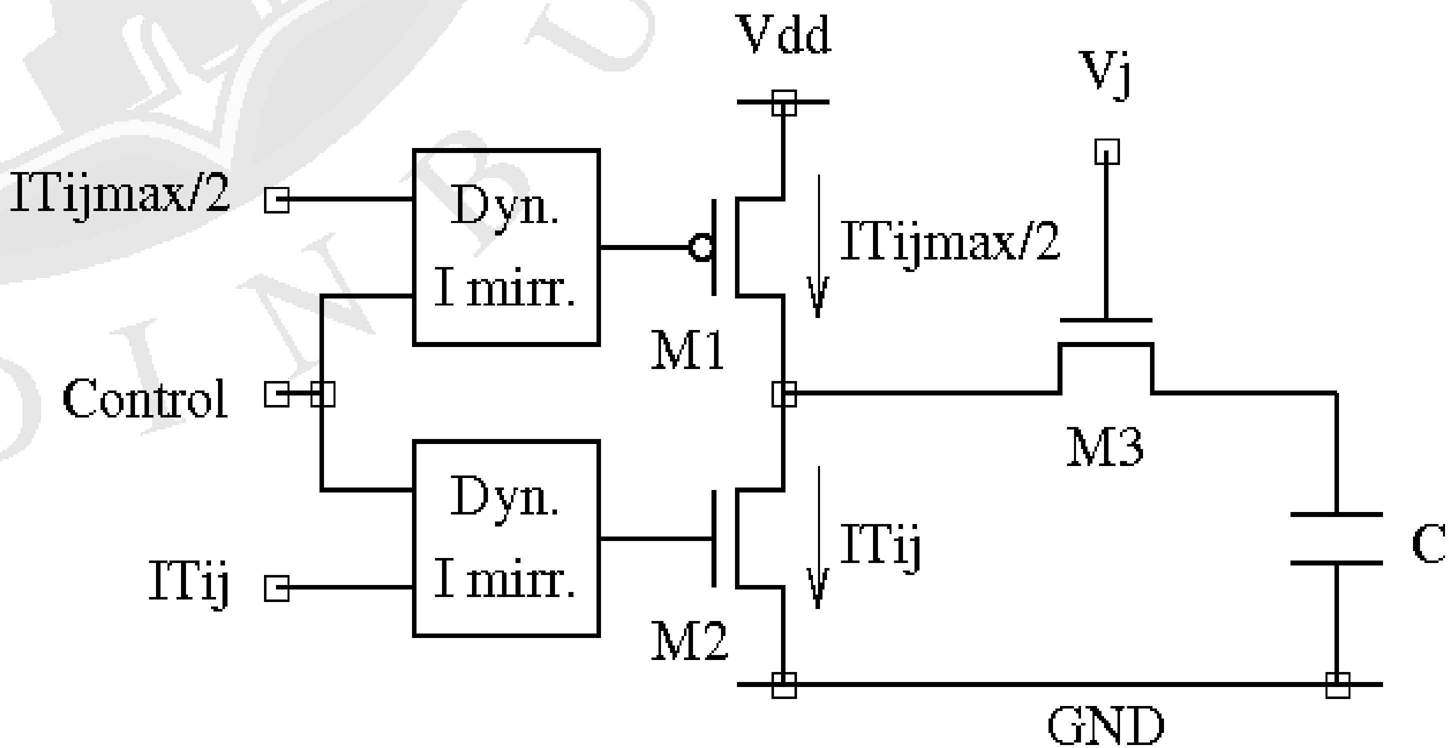
Implementing filters

- Analogue filters may be implemented using interconnected integrators with different scaling factors (*K*).



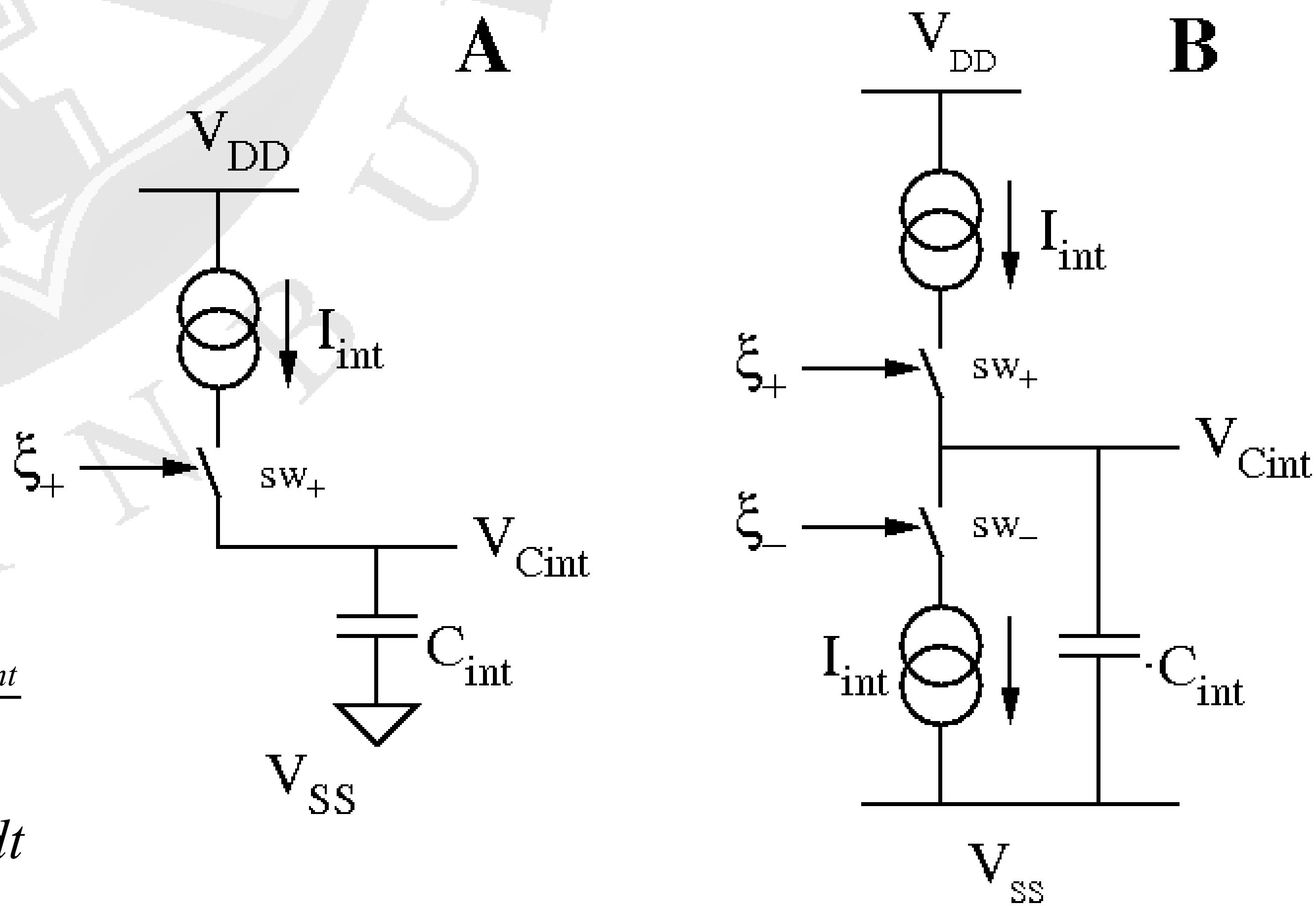
Portable synapse circuit

- Used in auditory neuromorphic signal processing
  - Essentially an integrator: input pulse, output voltage



Implementing integrators

- Adapt synapse model using PWM inputs ($\varepsilon_{+/-}$)

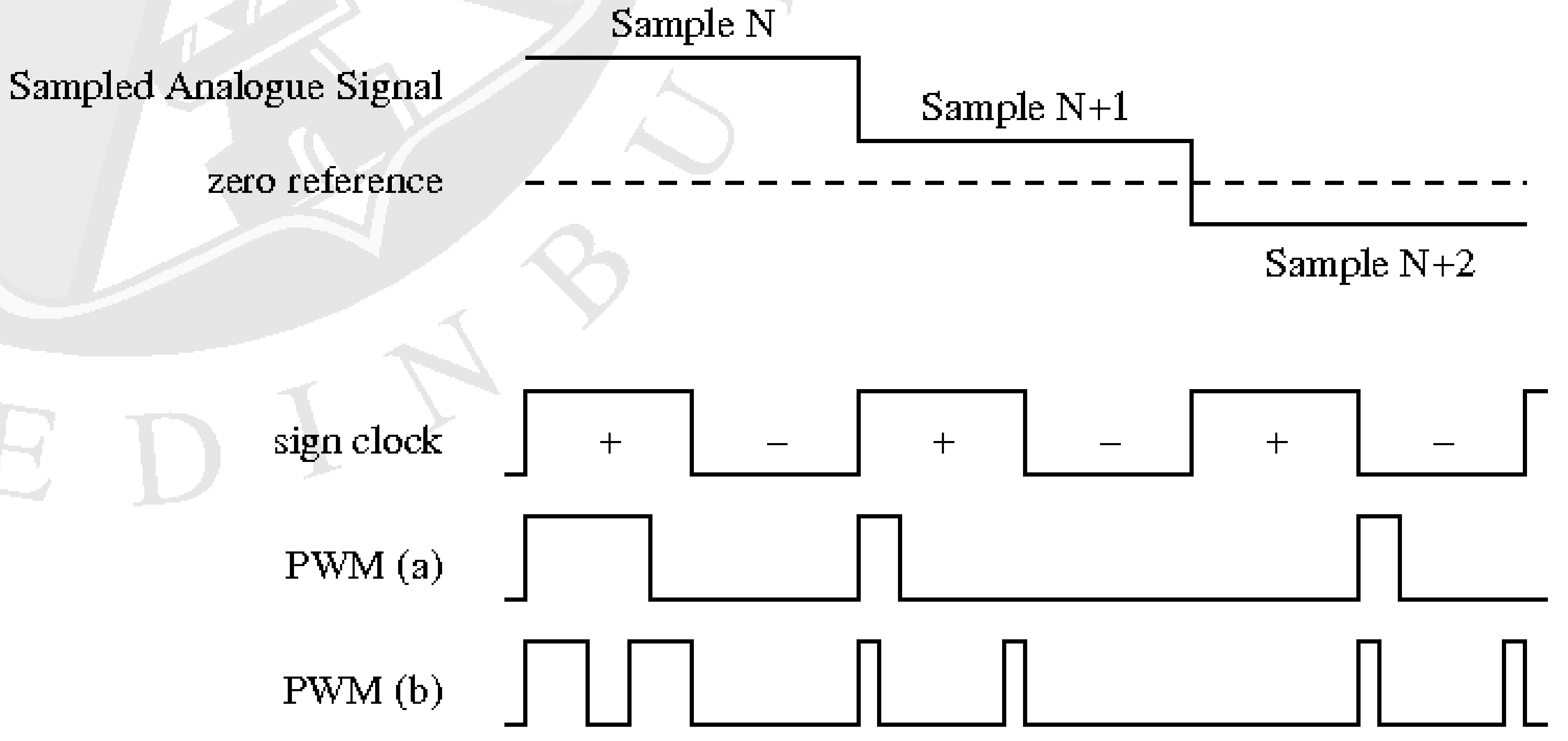


$$I_{int} = C \frac{dV_{Cint}}{dt}$$

$$dV_{Cint} = \frac{I_{int}}{C}\,dt$$

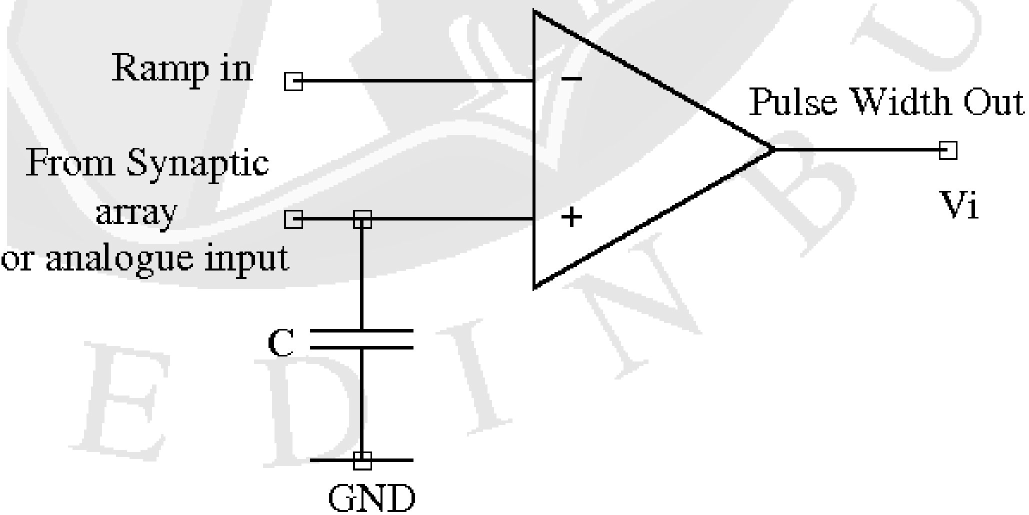
Adapt PWM coding scheme

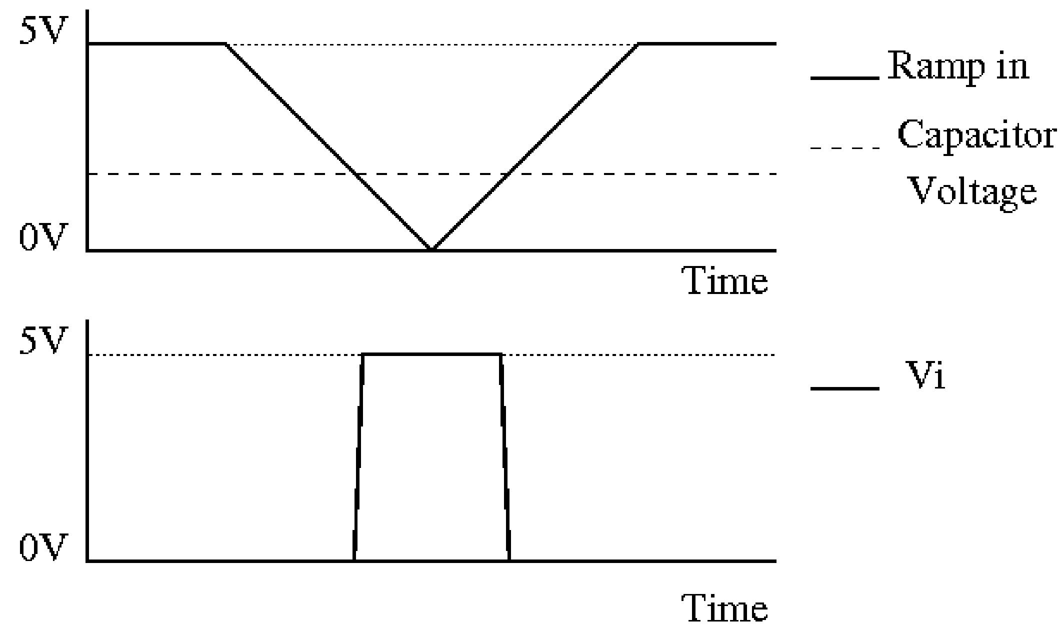
- To represent positive ***and*** negative magnitudes



Adapt pulse width neuron

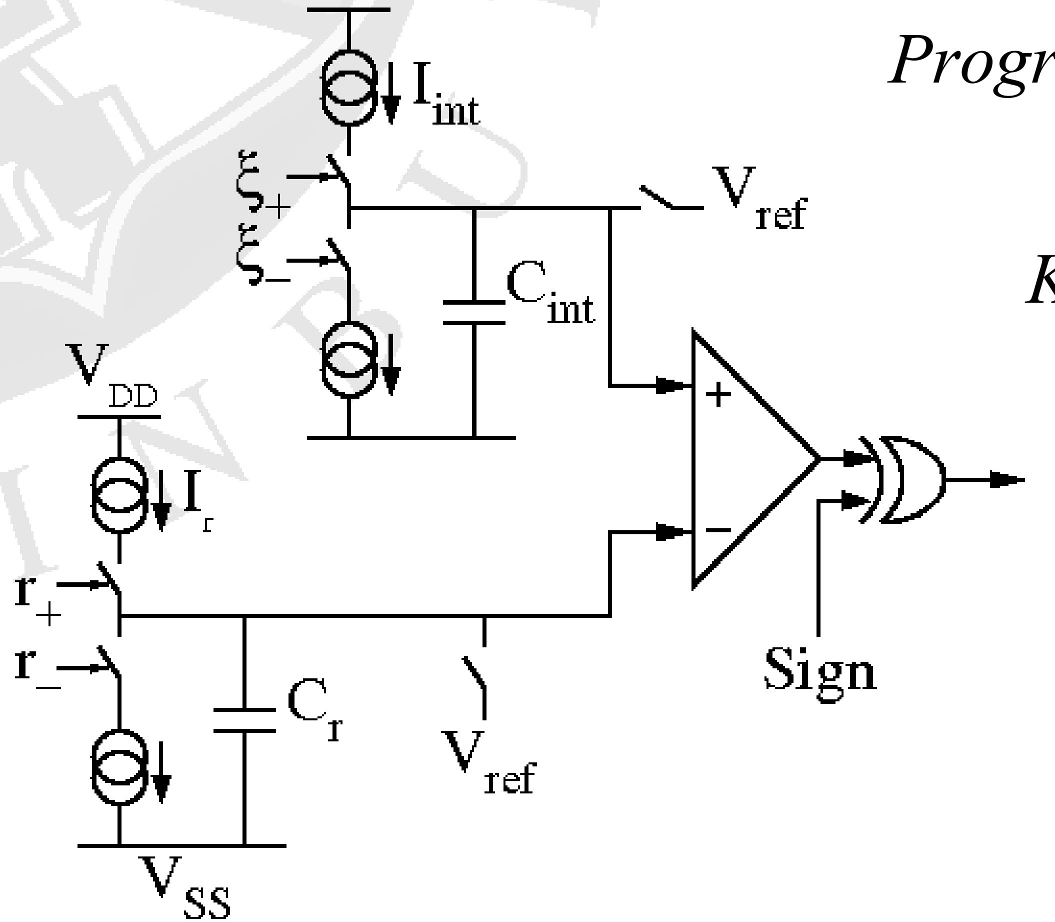
- Simple comparator fed with a (linear) ramp.





Palmo integrator

- Pulses in, pulses out, ***analogue*** inside.

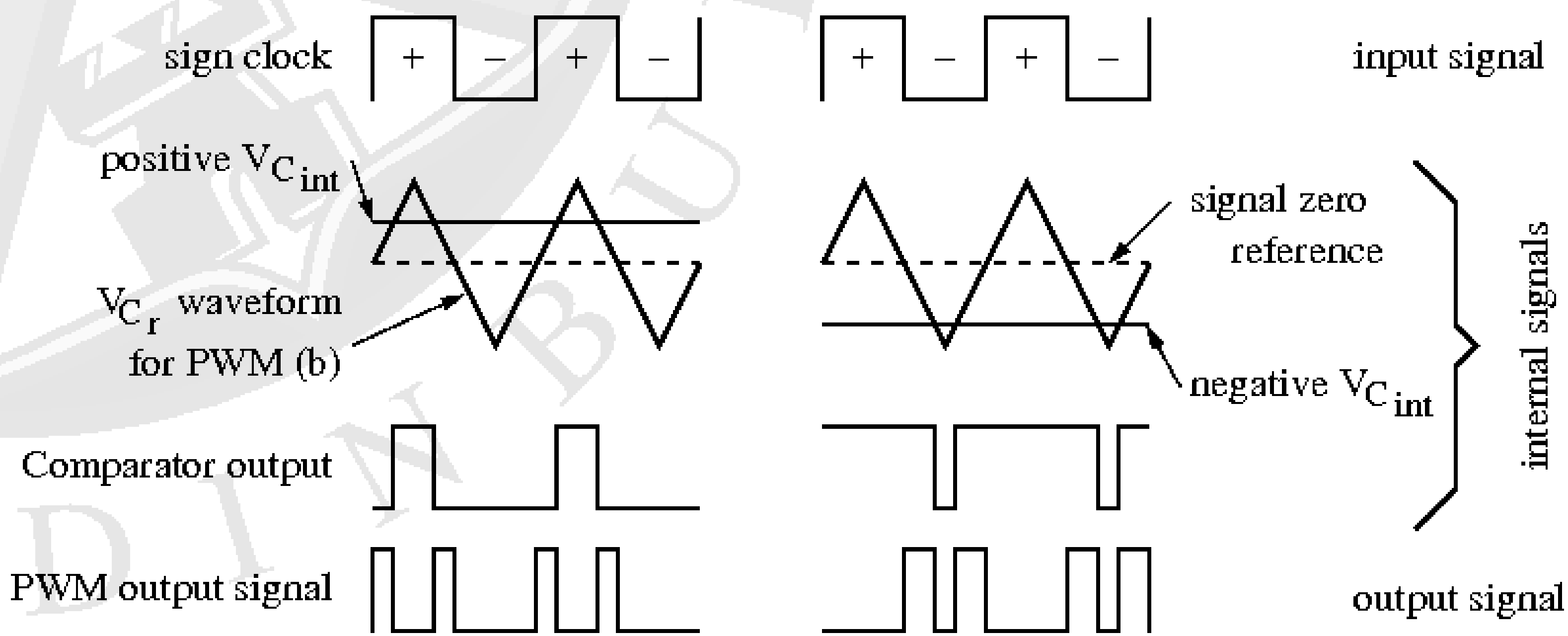


Programmable gain, K

$$K = \frac{C_r}{C_{\mathrm{int}}} \cdot \frac{I_{\mathrm{int}}}{I_r}$$

Palmo waveform diagrams

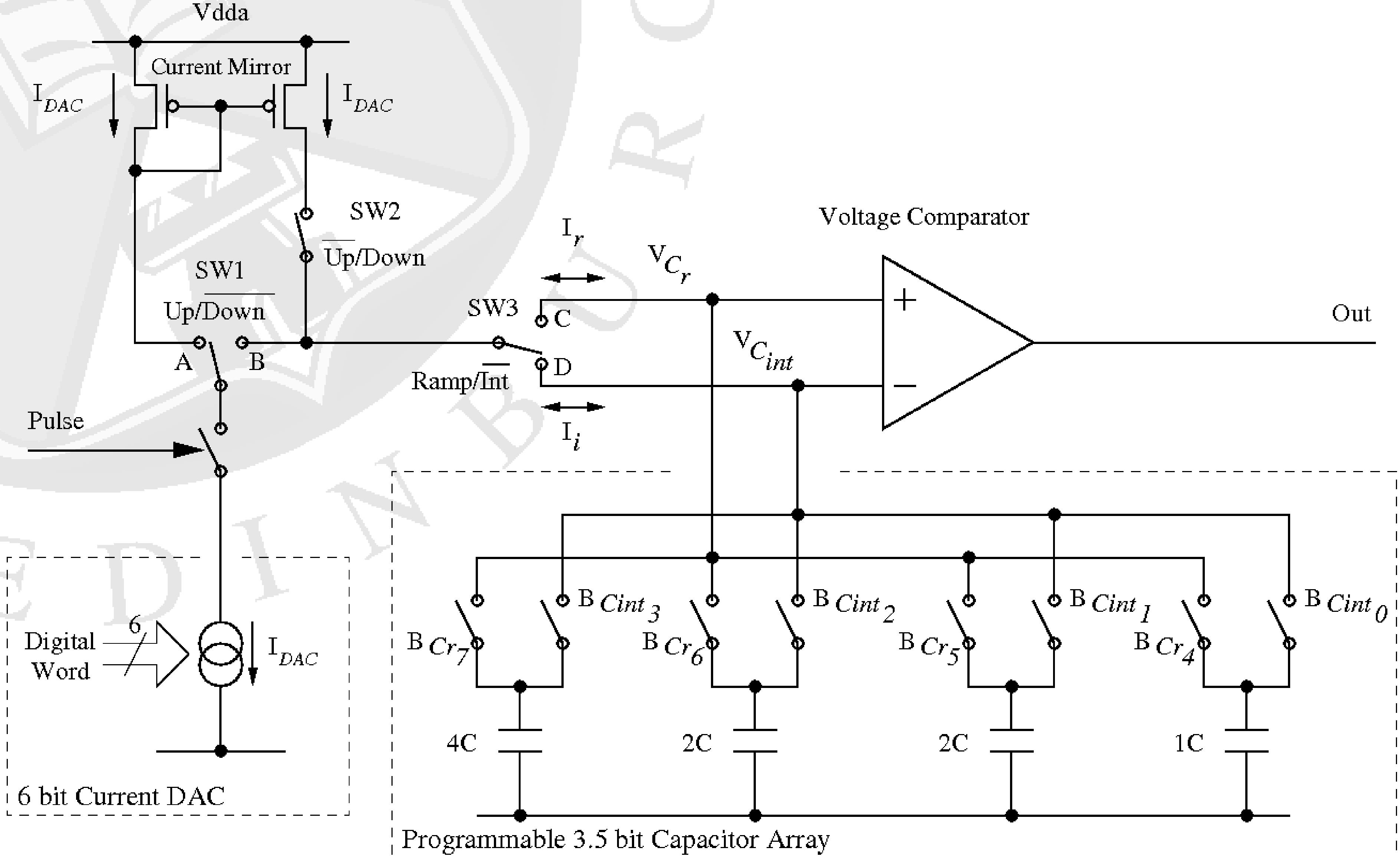
- Illustrate integrator concept



(a) positive value on signal integrator output

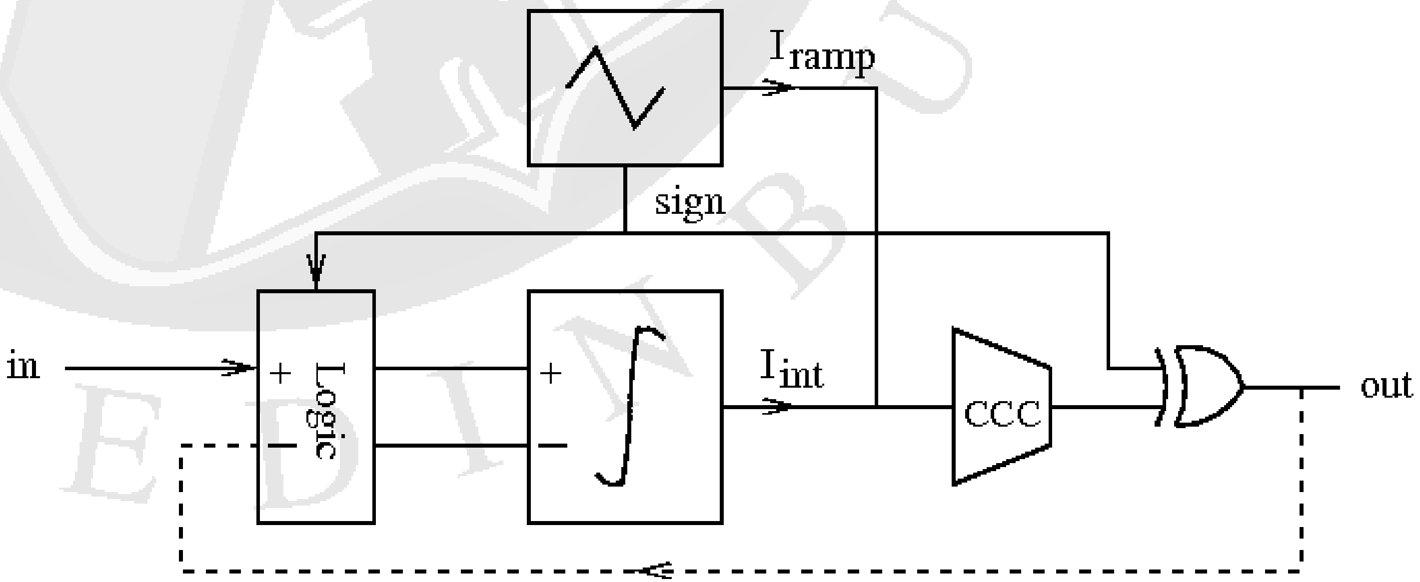
(b) negative value on signal integrator output

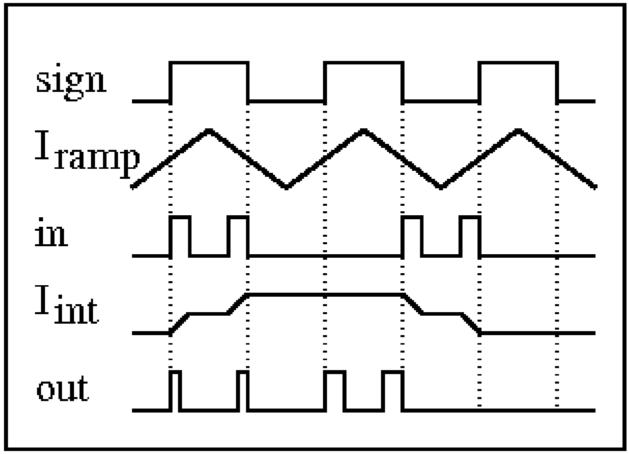
Palmo: voltage domain circuits



Palmo: current domain circuits

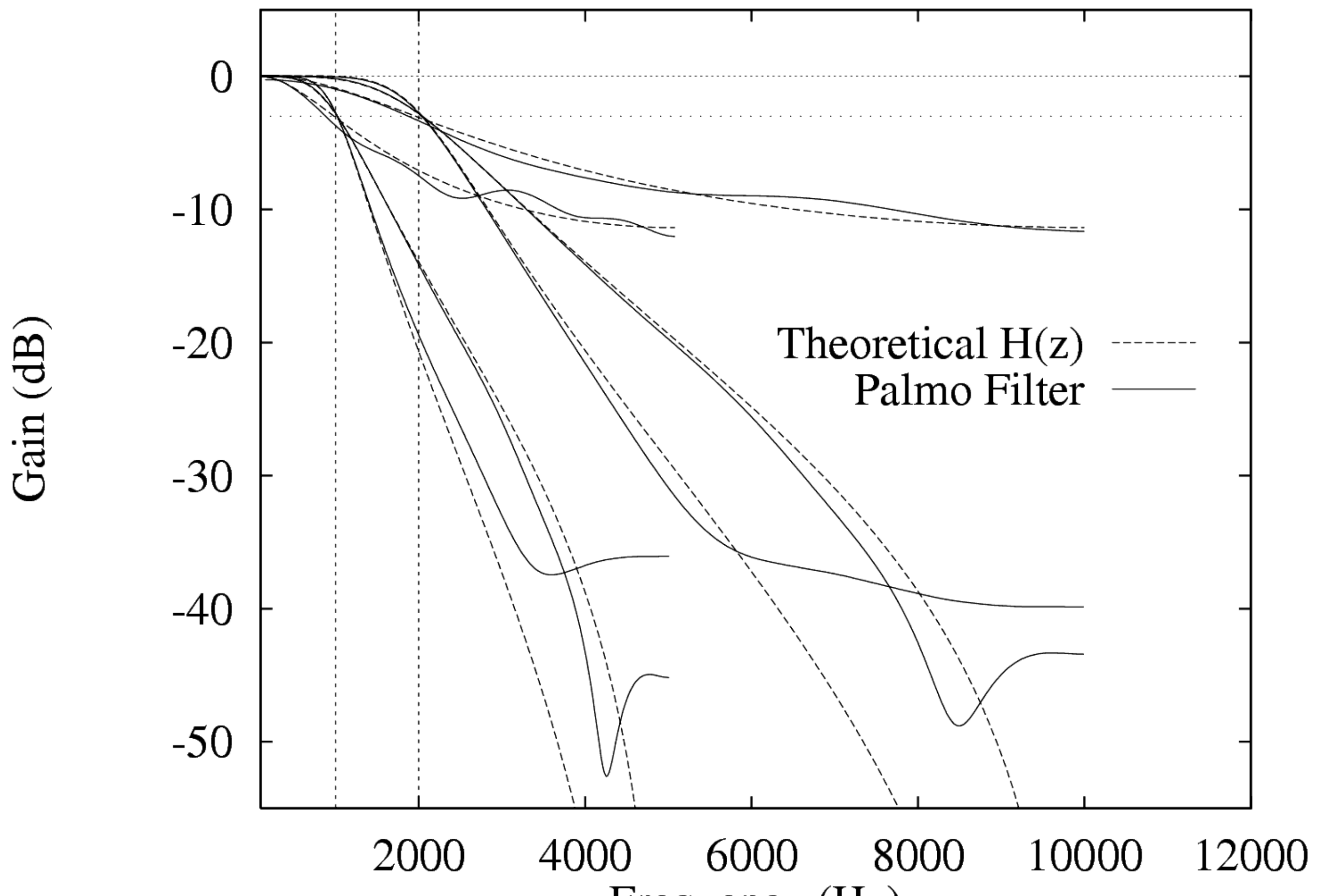
- Log domain integrator implemented





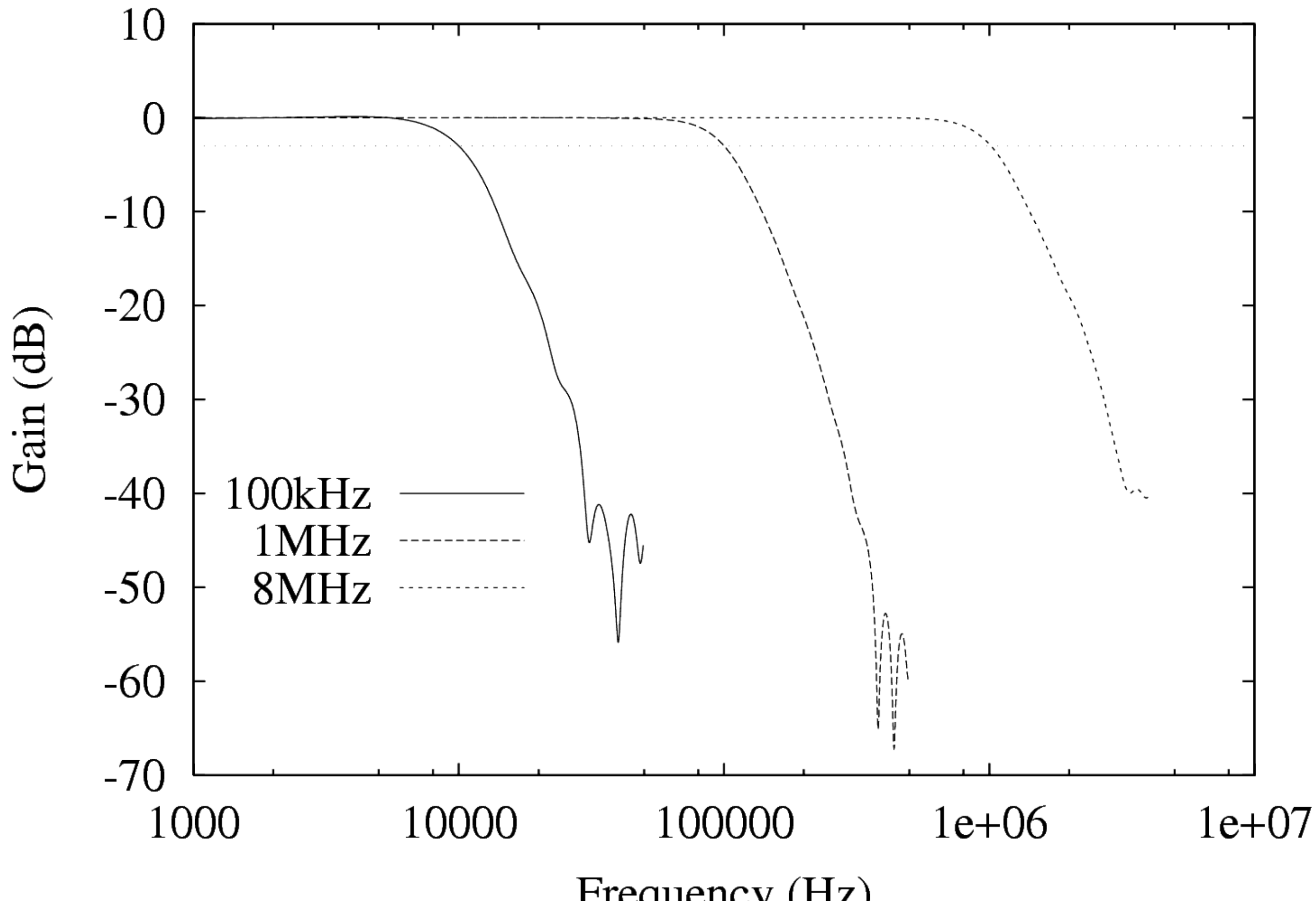
Palmo: voltage mode chip results

- 1st, 2nd & 3rd order Butterworth filters



Palmo: current mode chip results

- 3rd order Butterworth filters: 3 cut-off frequencies



Palmo: performance

- Voltage domain circuits offer limited SNR of 40 – 50 dB and limited sampling frequency 500 kHz.
  - Mirroring inaccuracies, limited supply headroom, slow voltage mode comparators.
- Current mode circuits performed best at sampling frequencies of around 1 MHz where SNR of over 60dB has been attained.
- Palmo equivalence to switched capacitor miller integrator demonstrated [key ref 1.]

Palmo: features

- Programmable analogue integrator cell
  - Fully programmable voltage mode, current mode and log domain implementations
- Pulse width signals used for communicating analogue signals between cells
  - Analogue processing within cells.
  - PWM signals synchronised to a sign clock.
- Time encoding of analogue information in digital PWM – easy programmable cell interconnect.

Programmable architecture #2

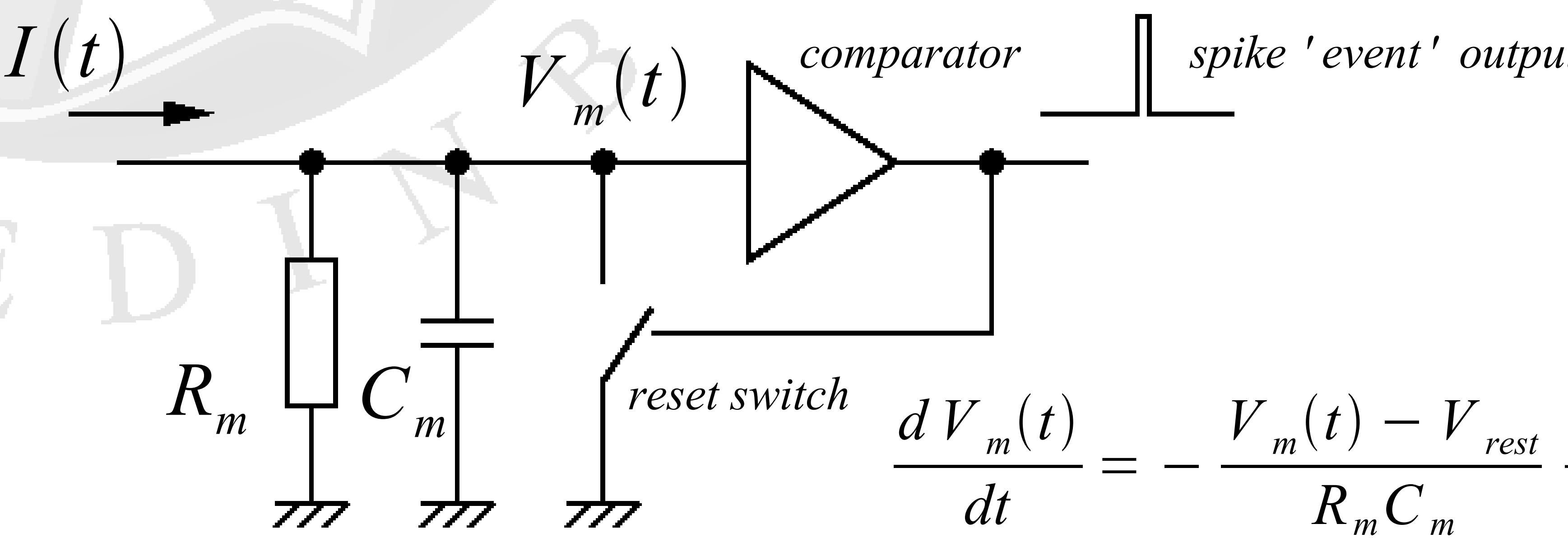
Event coding

Changing the pulse coding scheme

- Neural network implementations and ***Palmo*** use pulse based coding techniques.
  - predominantly pulse width modulation schemes
- Neither make any use of the ***relative time of occurrence*** of pulses.
- More biologically plausible neuromorphic systems take advantage of ***pulse*** or ***spike timing***.
  - integrate and fire neuron model, spike time dependent weight adaption etc.

Event coding: an example

- The ***integrate and fire*** neuron model
  - information conveyed by ***discretely occurring events*** and ***time intervals*** that separate them.



$$\frac{dV_m(t)}{dt} = -\frac{V_m(t) - V_{rest}}{R_m C_m} + \frac{I_{(t)}}{C_m}$$

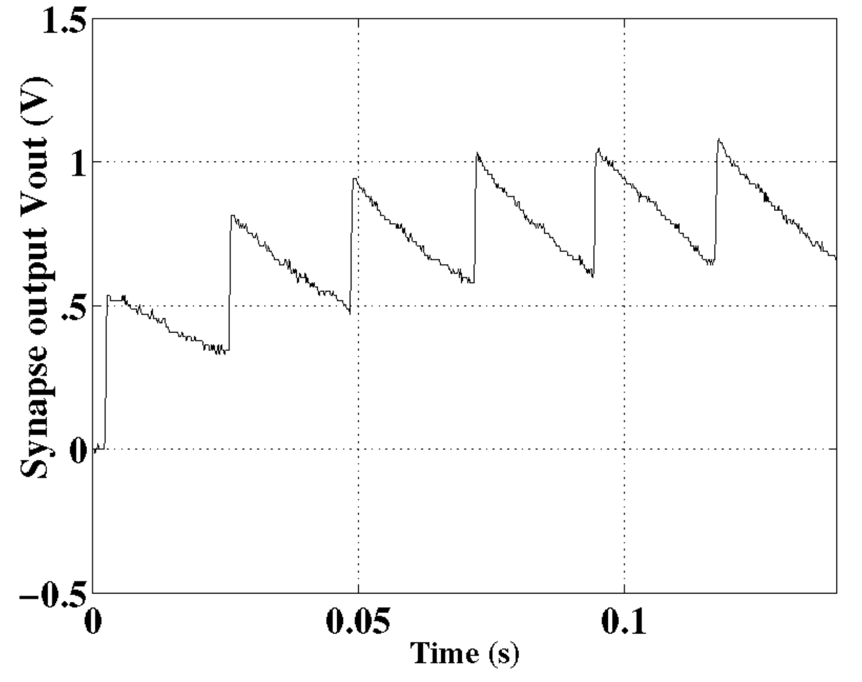
Time dependent synapse function

- The ***exponential summing*** synapse

$$i_{BA}(t) = \Theta(t)\,\omega_{BA}\,\mathrm{e}^{\frac{-t}{\tau_d}}$$

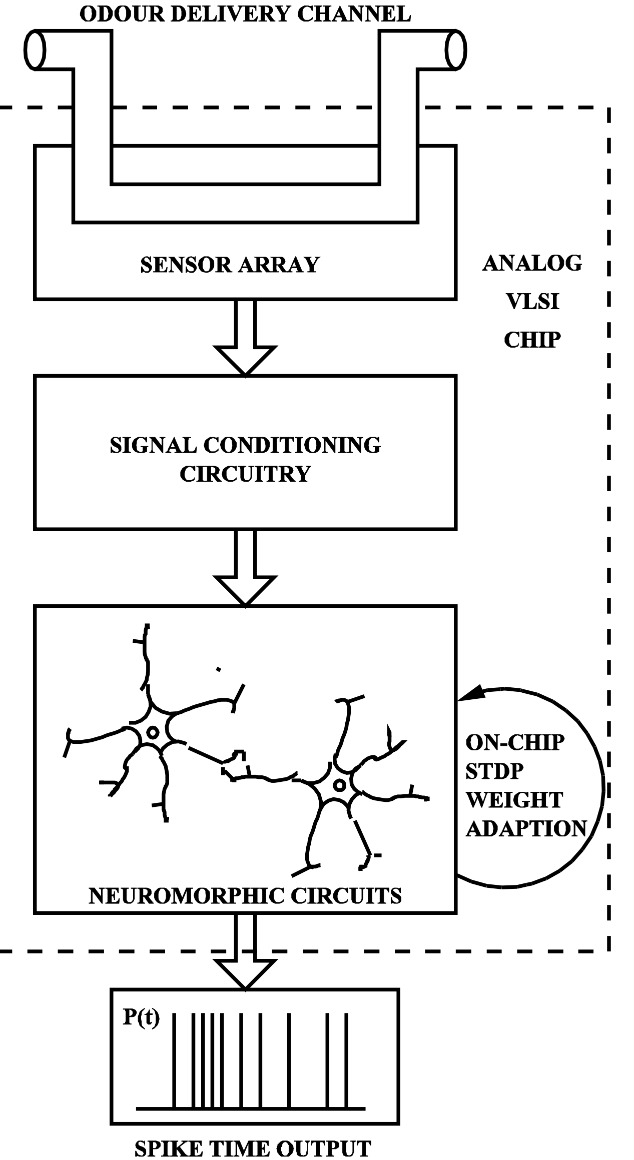
$$I_{BA}(t) = \sum_n i_{BA}(t - t_n)$$

- Synapse output
  - Pre-synaptic spike event period = 23mS, V_{wt} = 500mV.



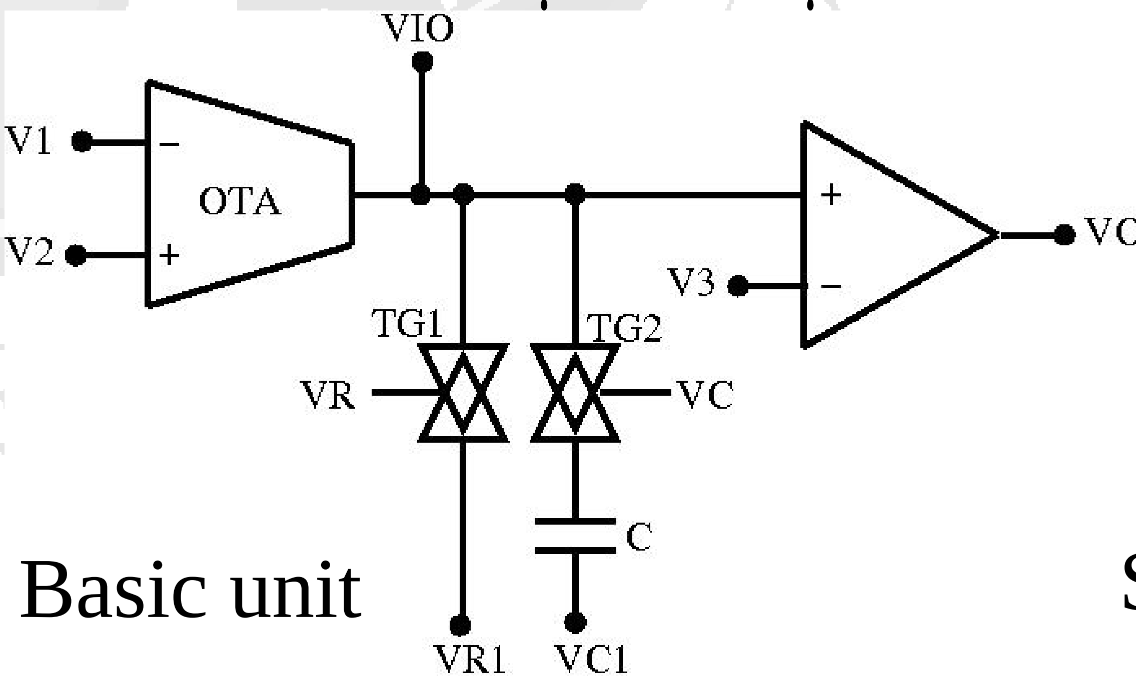
Electronic nose project

- Neuromorphic analogue VLSI
  - integrate and fire neurons
  - exponential summing synapses
  - with weight adaption
- Can we implement all the neuromorphic circuits used in this project?
  - Using just one programmable analogue cell?



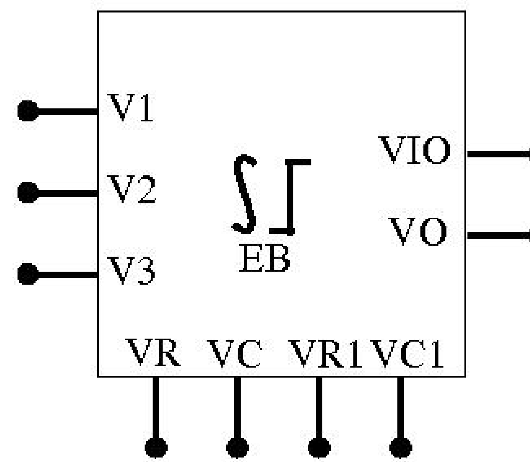
Event coding programmable architecture: basic unit

- Derived from circuits designed to implement the electronic nose neuromorphic architecture
  - EB size 190μm x 150μm in AMS 0.35μmm CMOS



Basic unit

(a)

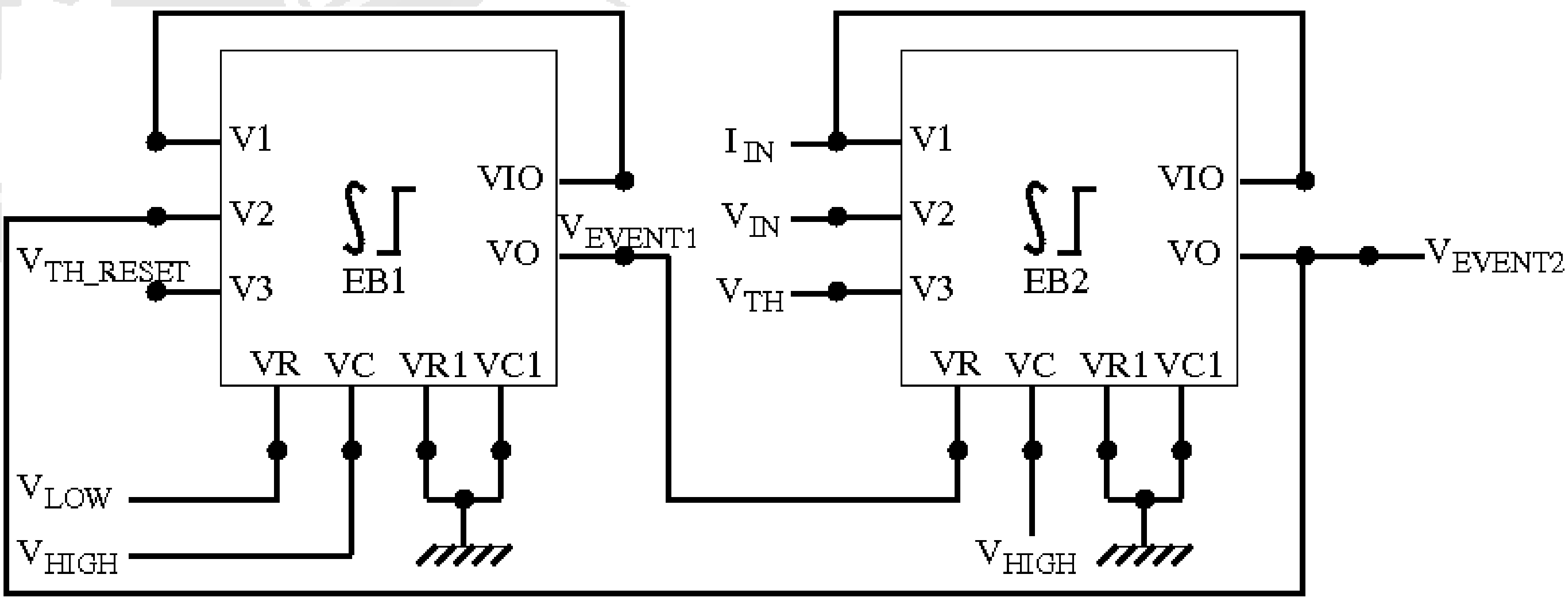


Symbol

(b)

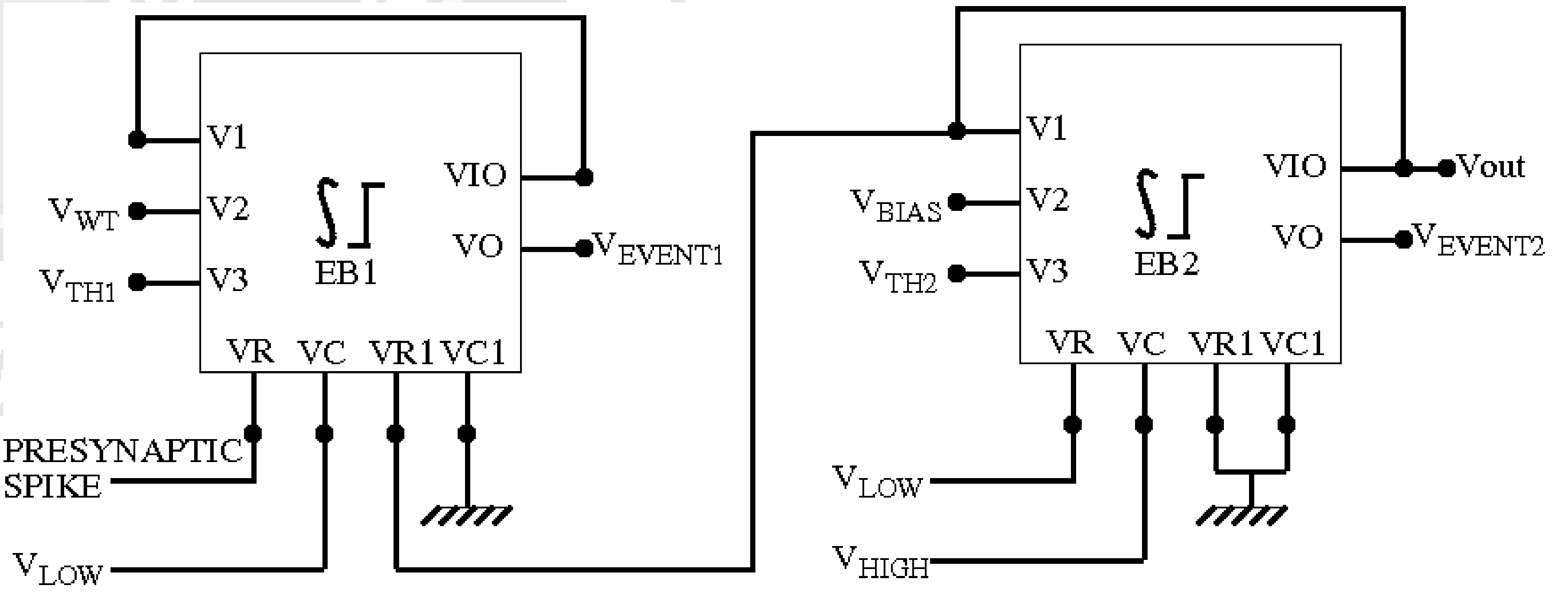
Integrate and fire neuron

- Implemented by connecting 2 basic units



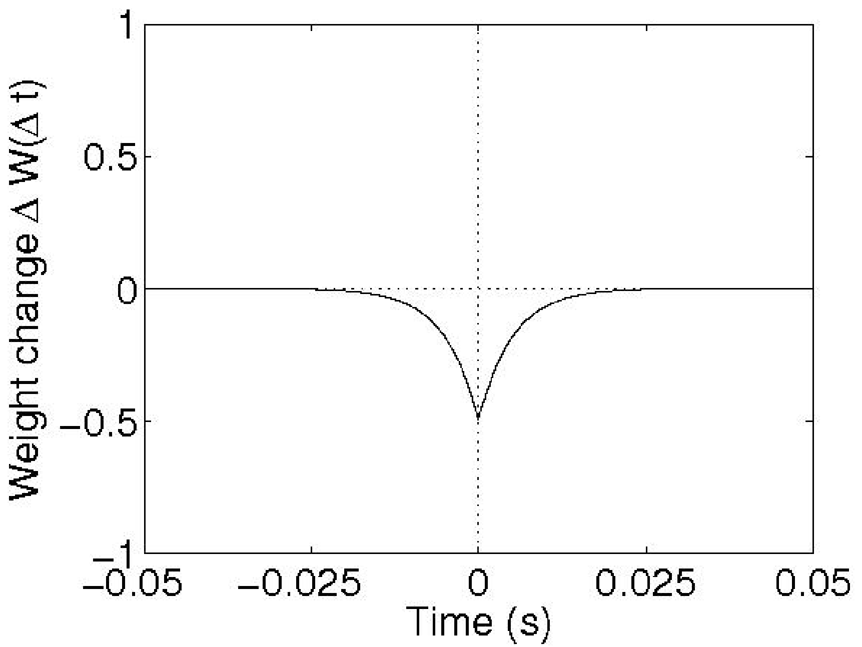
Exponential summing synapse

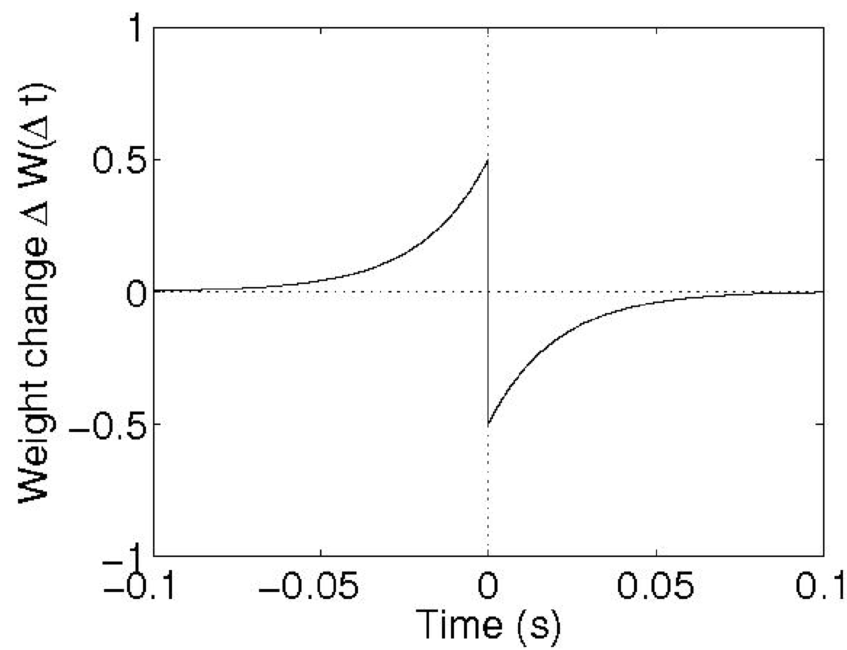
- Implemented by connecting 2 basic units



Spike time dependent weight adaption

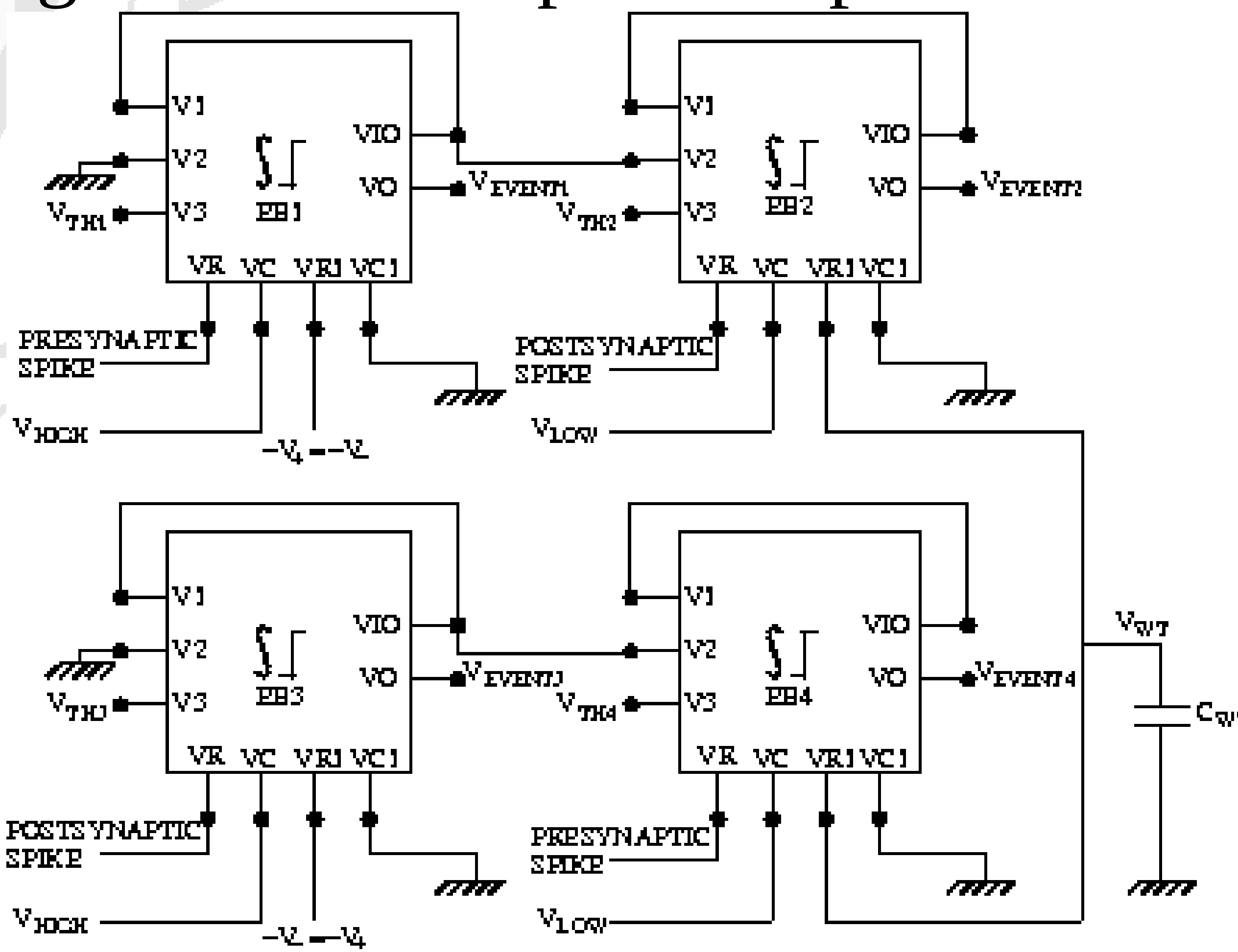
- Egger et al. 1999
- Song et al. 2000
  - Synaptic weight change based on ***pre*** and ***post*** synaptic ***spike time correlation***





Weight adaption circuit #1 (Egger '99)

- Implemented using 4 basic units plus a capacitor

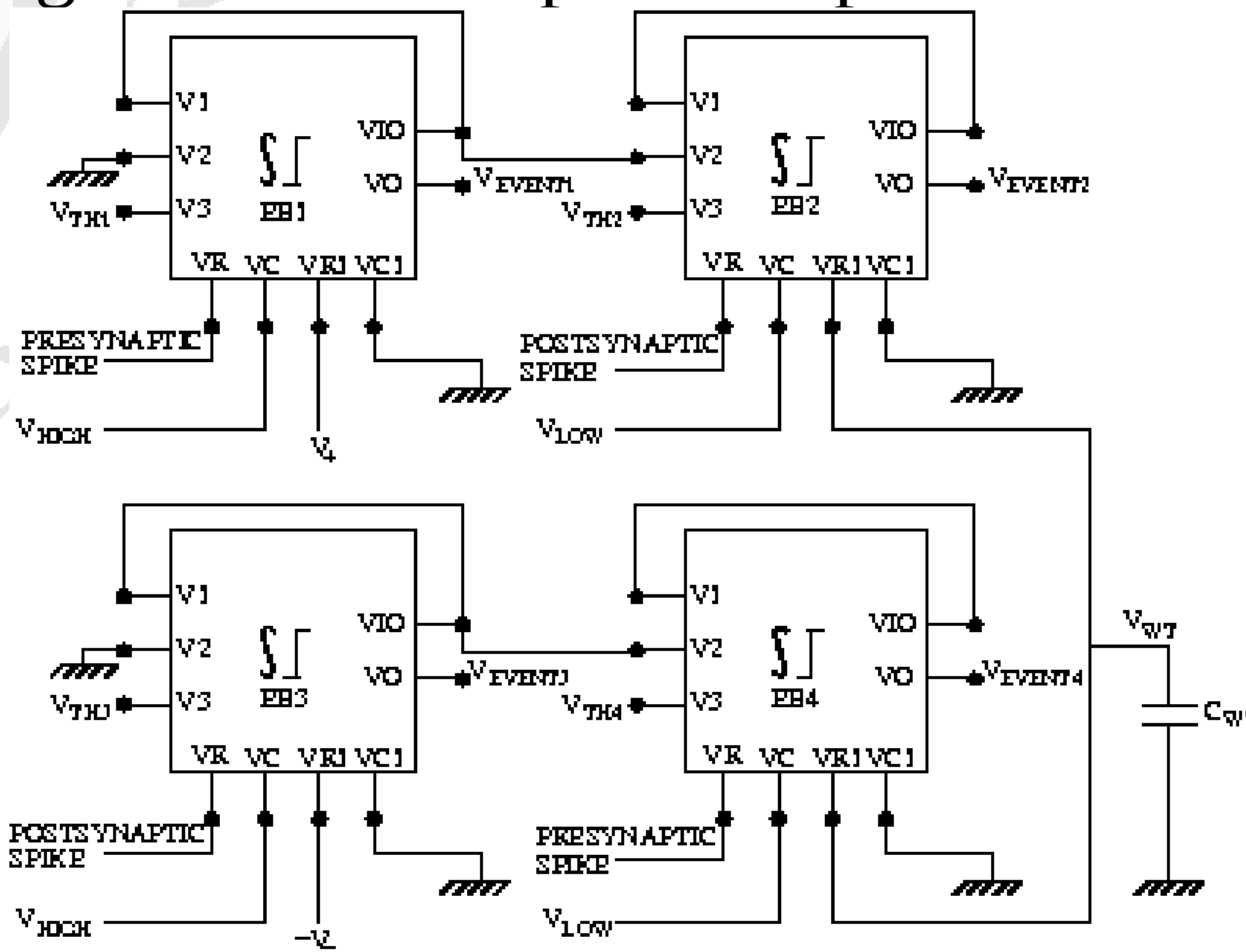


$$\Delta W(\Delta t) = A\, e^{\frac{-\Delta t}{t_{1n+}}} \quad if\ \Delta t > 0$$

$$\Delta W(\Delta t) = A\, e^{\frac{\Delta t}{t_{1n-}}} \quad if\ \Delta t \leq 0$$

Weight adaption circuit #2 (Song '00)

- Implemented using 4 basic units plus a capacitor

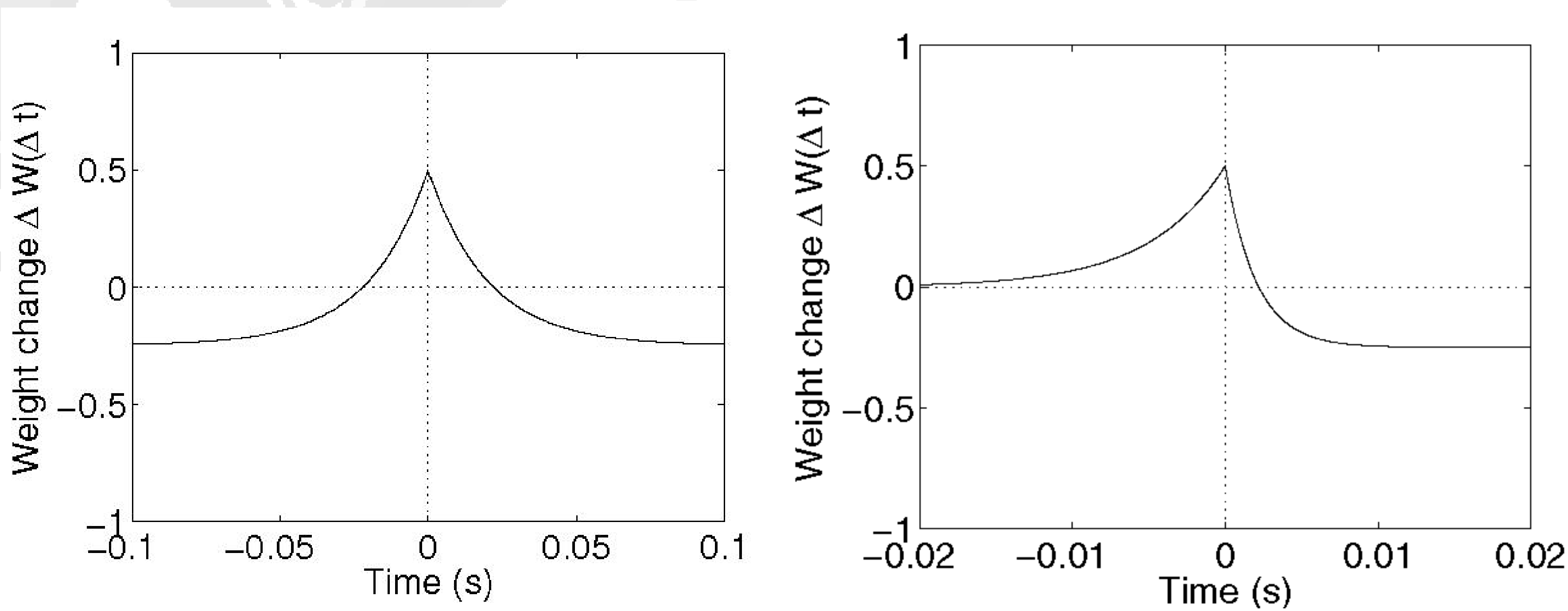


$$\Delta W(\Delta t) = A_{+} e^{\frac{-\Delta t}{t_{1n+}}} \quad if\ \Delta t > 0$$

$$\Delta W(\Delta t) = A_{-} e^{\frac{\Delta t}{t_{1n-}}} \quad if\ \Delta t \leq 0$$

Spike time dependent weight adaption

- Dan et al. 1992
- Gerstner et al. 1996
  - Synaptic weight change based on ***pre*** and ***post*** synaptic ***spike time correlation***

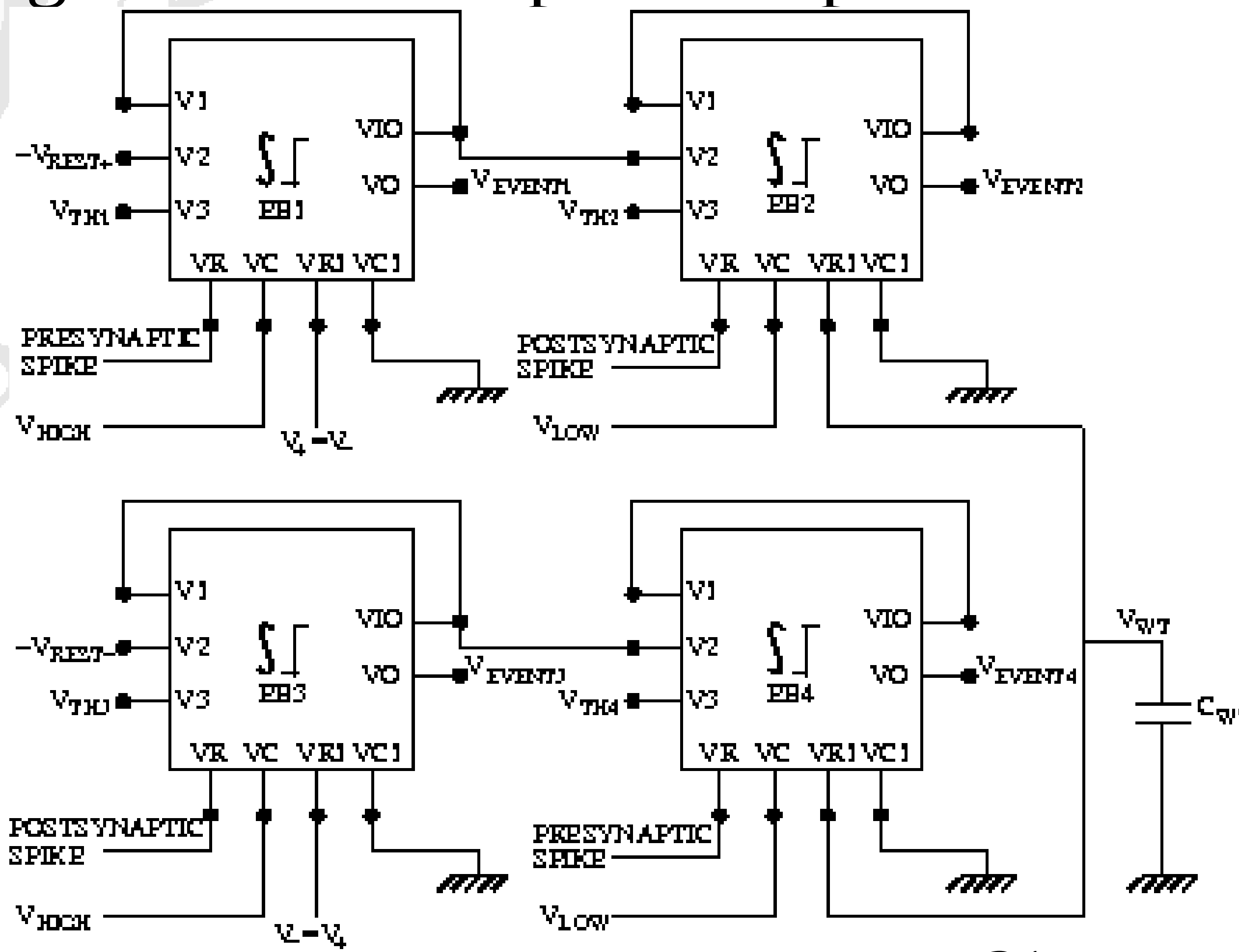


Weight adaption circuit #3 (Dan '92)

- Implemented using 4 basic units plus a capacitor

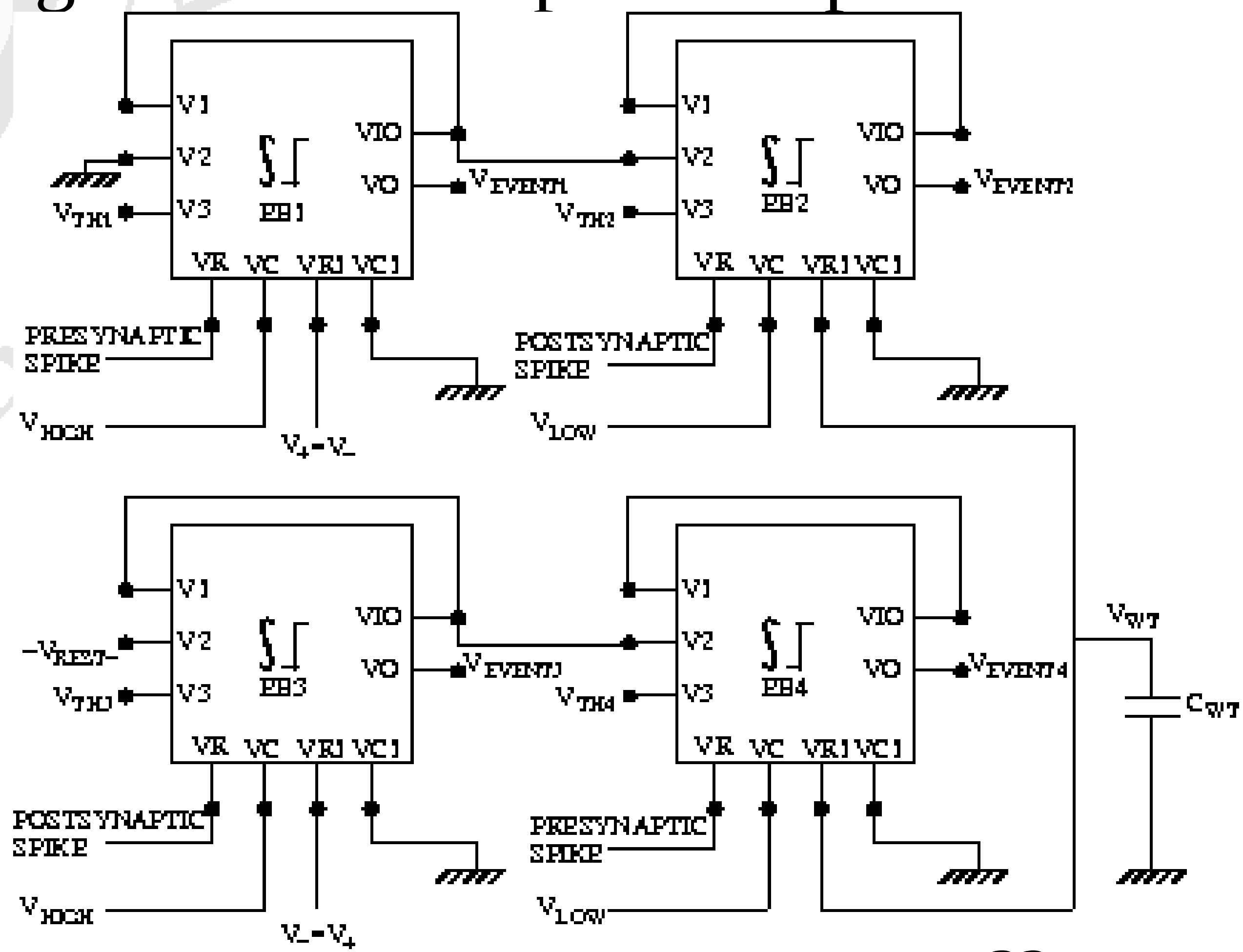
$$\Delta W(\Delta t) = (A_{+} - A_{rest+})\, e^{\frac{-\Delta t}{t_{1n+}}} + A_{rest+} \quad if\ \Delta t > 0$$

$$\Delta W(\Delta t) = (A_{-} - A_{rest-})\, e^{\frac{\Delta t}{t_{1n-}}} + A_{rest-} \quad if\ \Delta t \le 0$$



Weight adaption cct #4 (Gerstner '96)

- Implemented using 4 basic units plus a capacitor

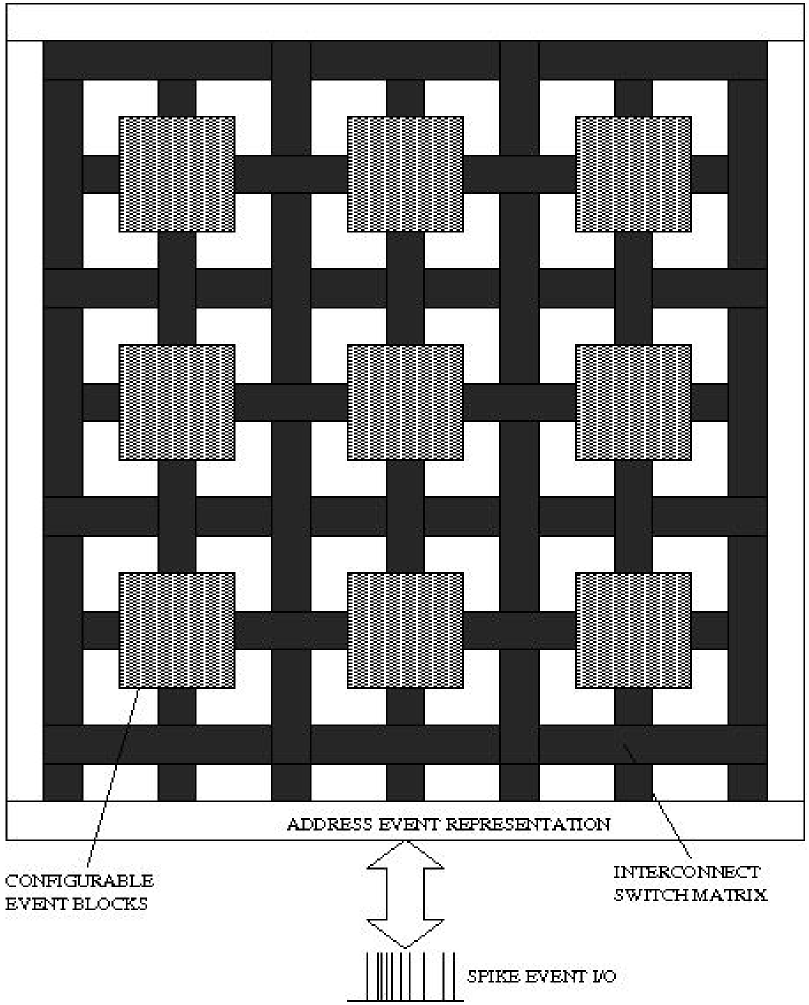


$$\Delta W(\Delta t) = (A_{+} - A_{rest+})\, e^{\frac{-\Delta t}{t_{1n+}}} + A_{rest+} \quad if\ \Delta t > 0$$

$$\Delta W(\Delta t) = (A_{-} - A_{rest-})\, e^{\frac{\Delta t}{t_{1n-}}} + A_{rest-} \quad if\ \Delta t \leq 0$$

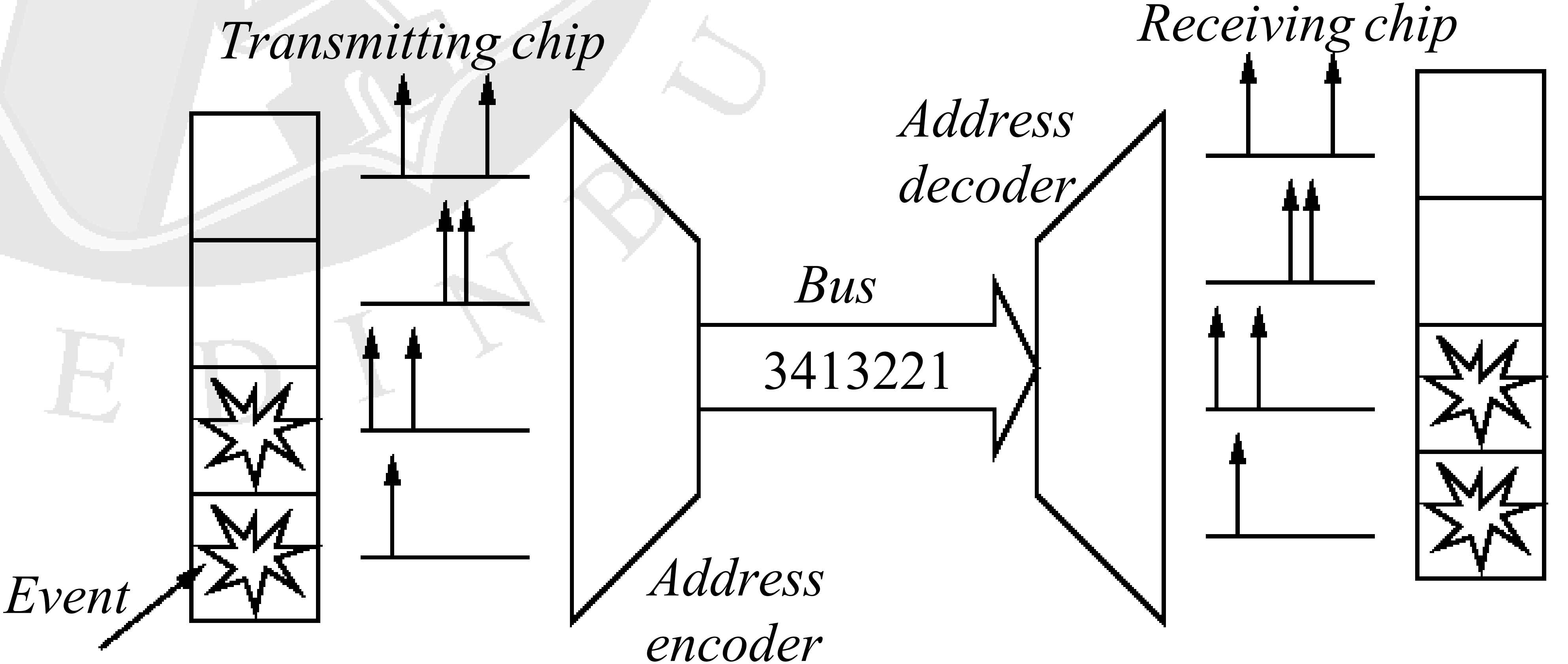
Event coded programmable array

- Array of basic units on chip.
- Programmable interconnect
- Programmable functionality
- Standard external AER interface



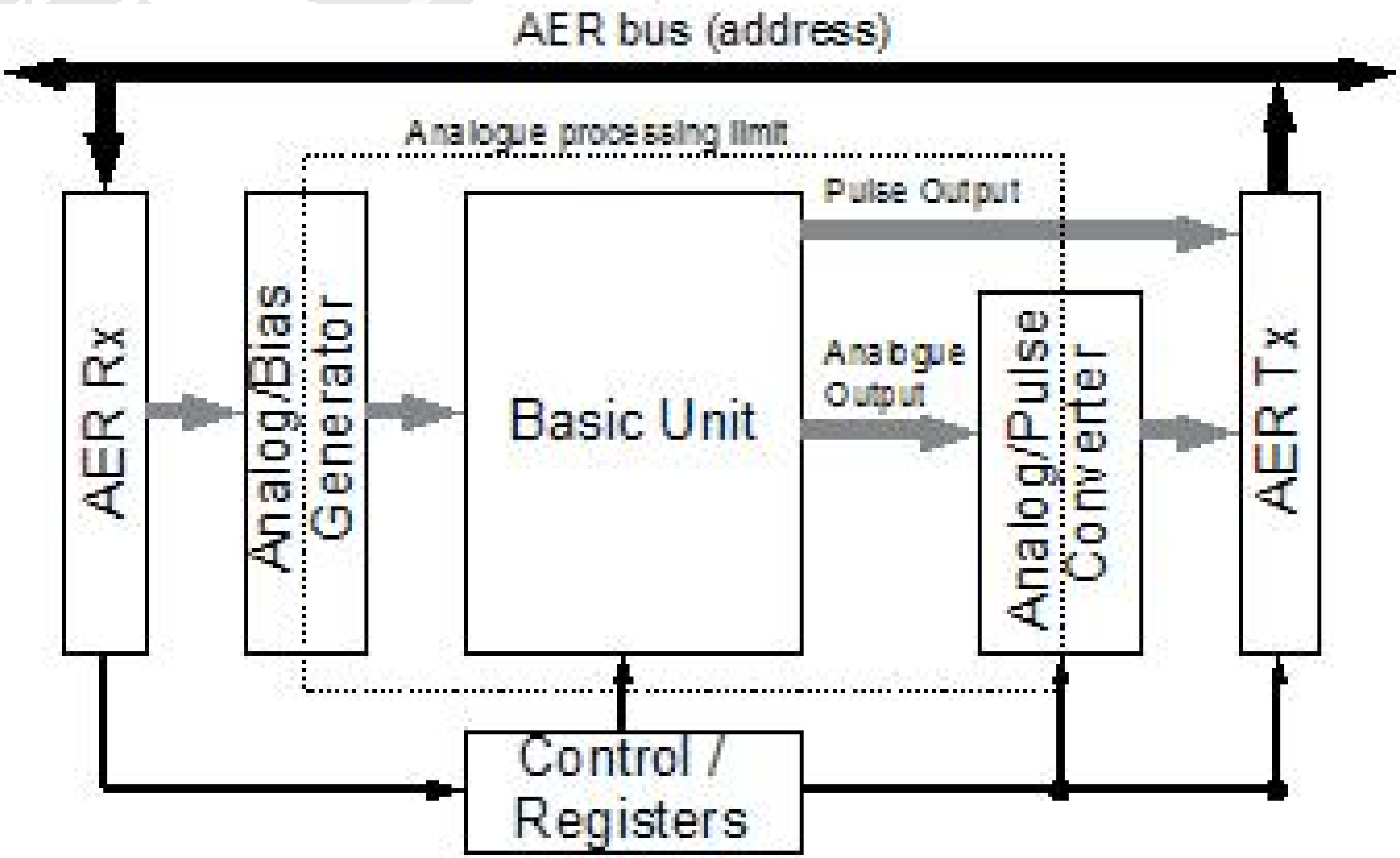
AER: Address Event Representation

- A communications protocol for connecting together chips containing spiking neuromorphs



Basic event coded building block

- Basic unit embedded in AER wrapper

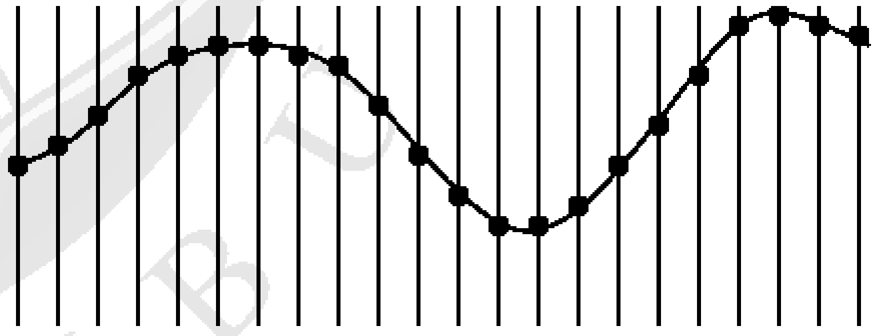


Issues in communication between event coded blocks

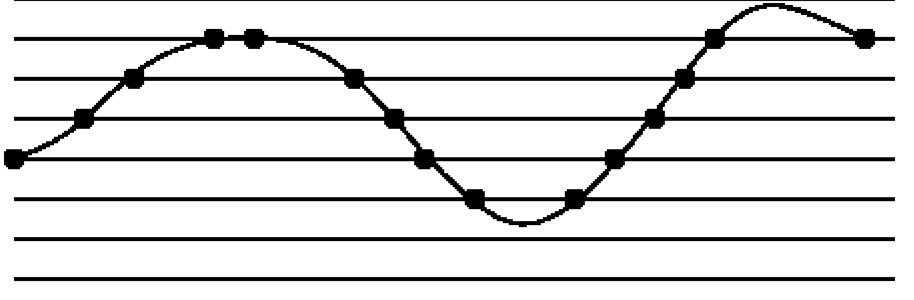
- In most topologies, blocks communicate using a digital or spike representation.
  - use AER.
- Some topologies connect ***analogue*** signals between event coded blocks.
  - associated problems using programmable analogue connections (switch matrix, drive capability, noise etc).
  - cluster several basic units within one AER wrapper?
  - convert analogue signals into an AER representation?

Implicit and Explicit sampling

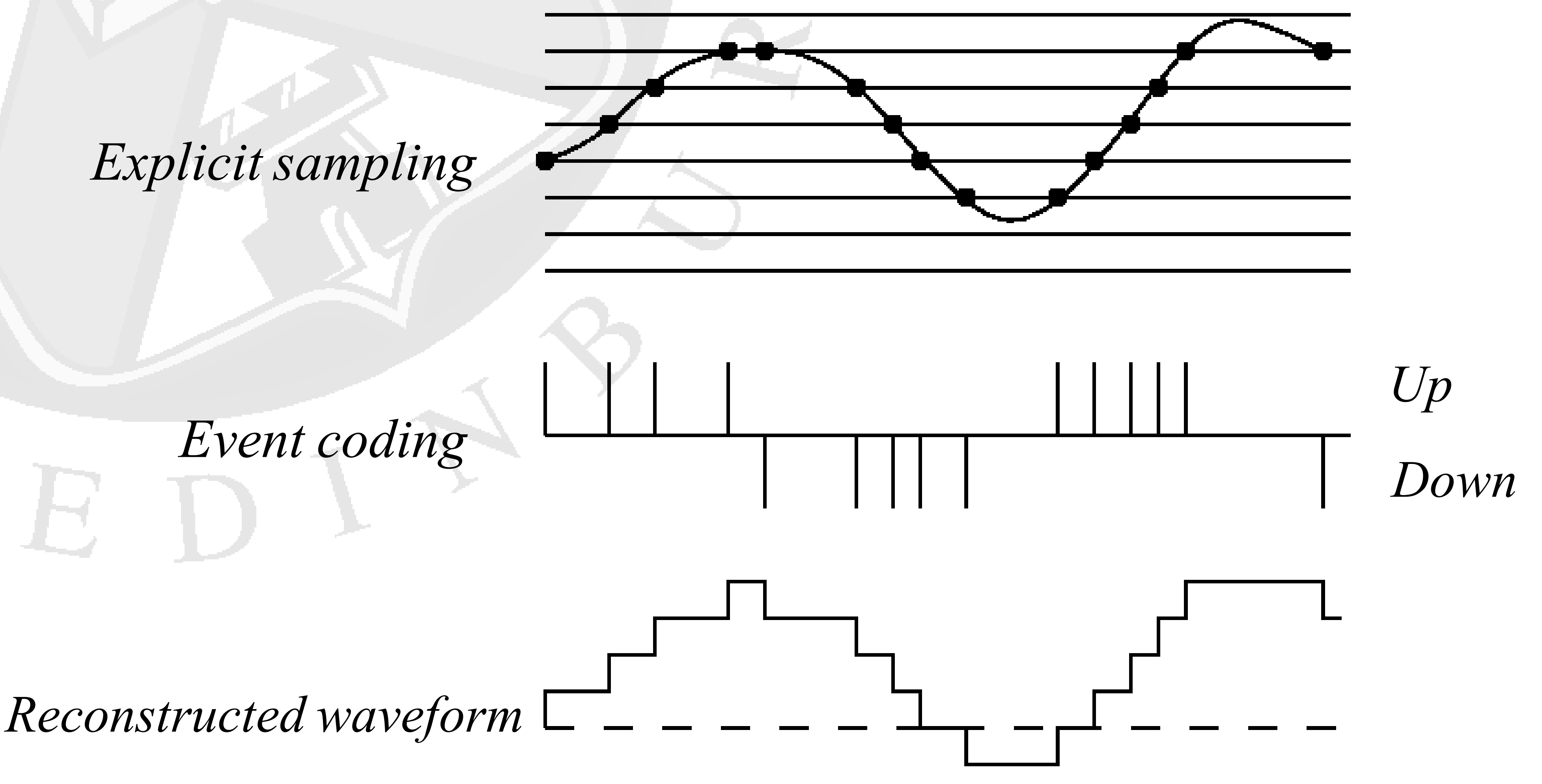
- Conventional (implicit) sampling of an analogue signal



- Explicit sampling of an analogue signal

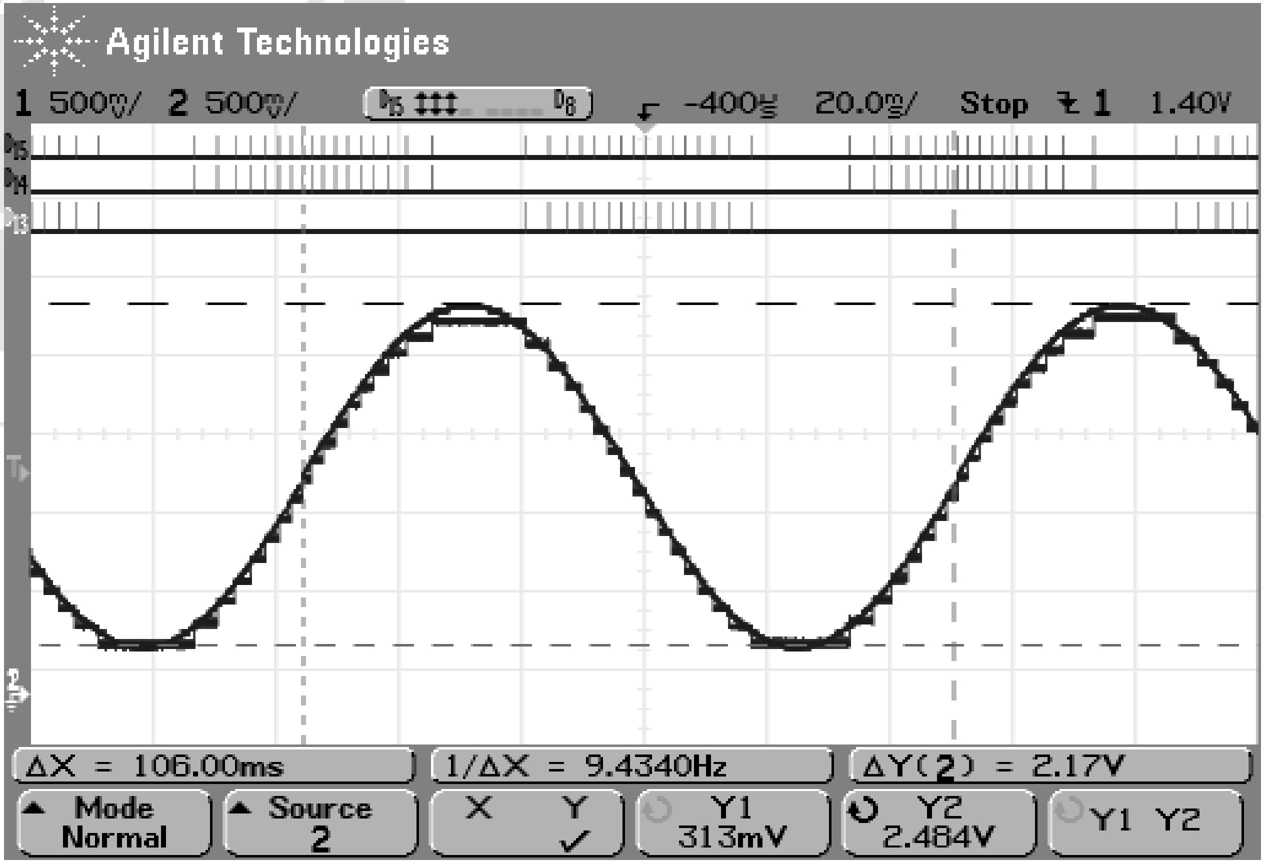


Explicit Sampling to Event Coding



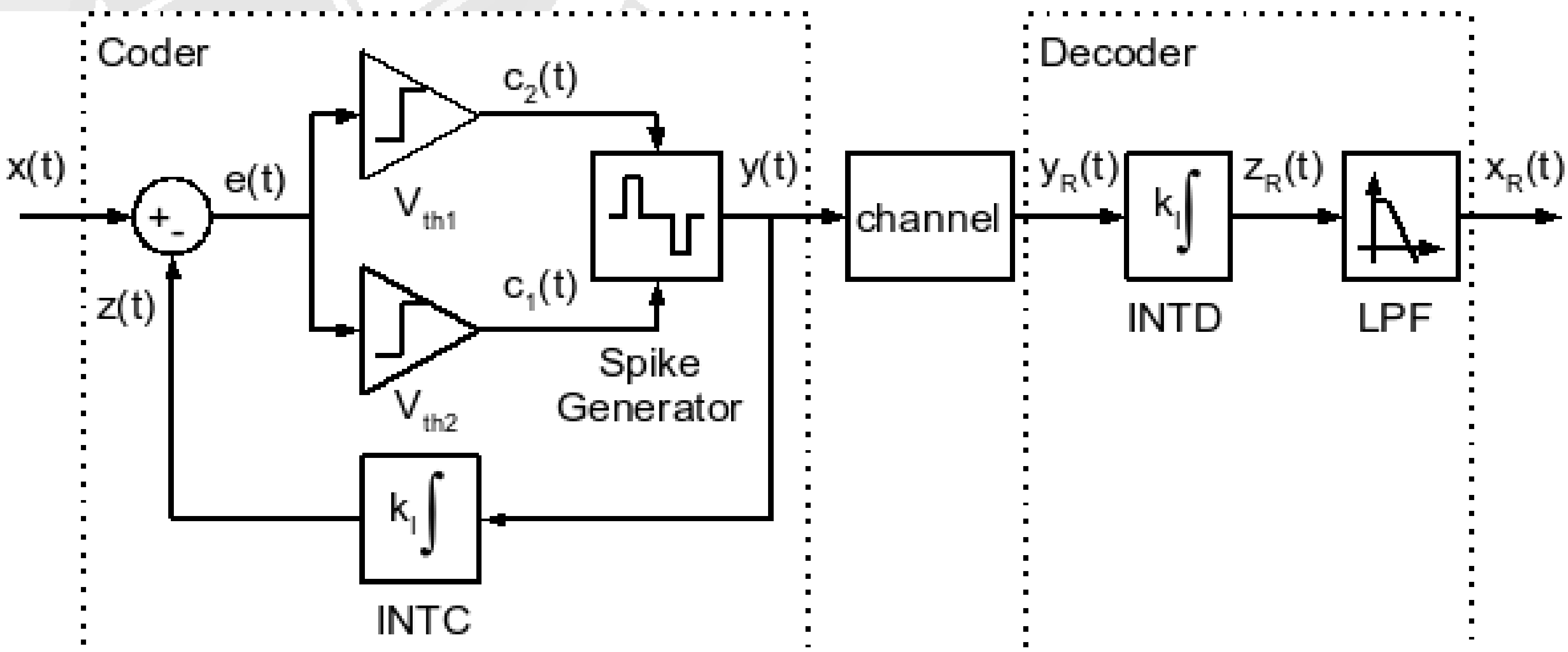
Spike event coding

- Explicit sampling of analogue waveform encoded as spike events and reconstructed waveform.



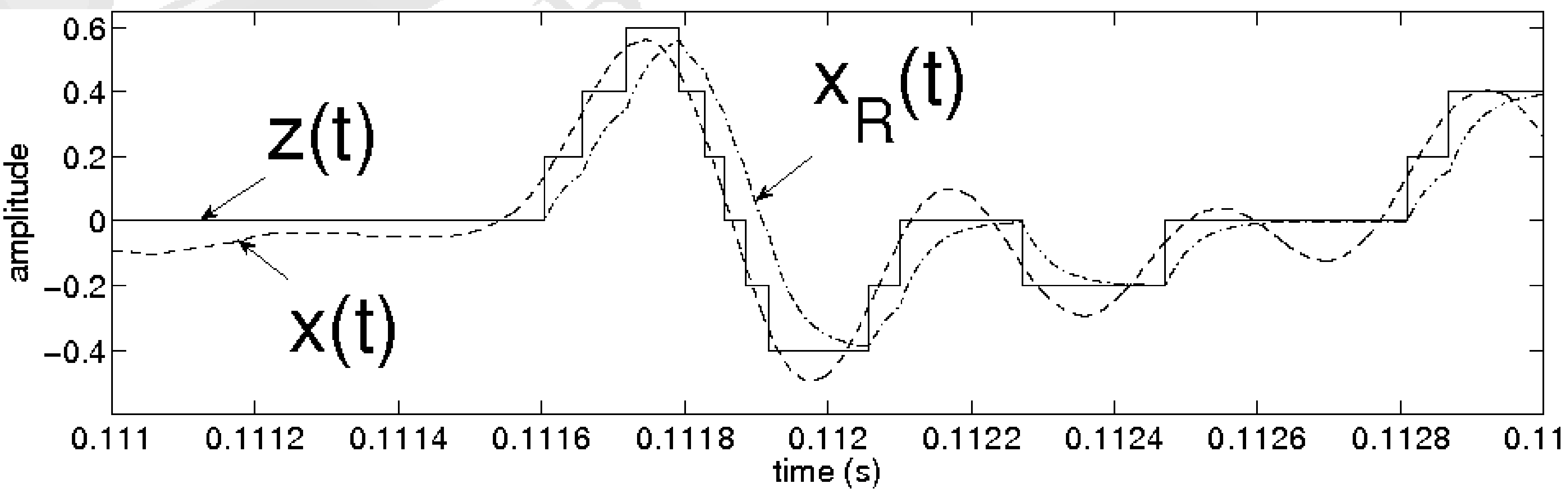
AER compatible analogue signals

- Spike event coding using ***asynchronous delta modulation***



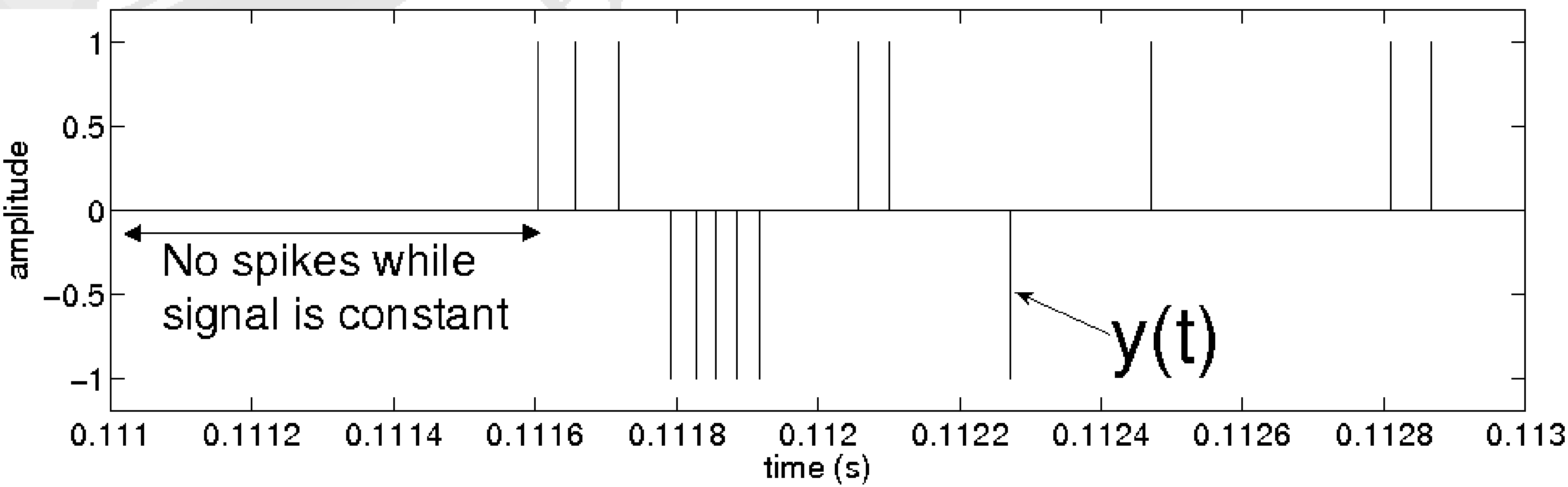
Speech signal

- Analogue signal used to illustrate ***asynchronous delta modulation*** coding/decoding principles
  - encoder analogue input signal, *x(t)*,
  - integrator output, *z(t)*, decoded output, $x_R(t)$.



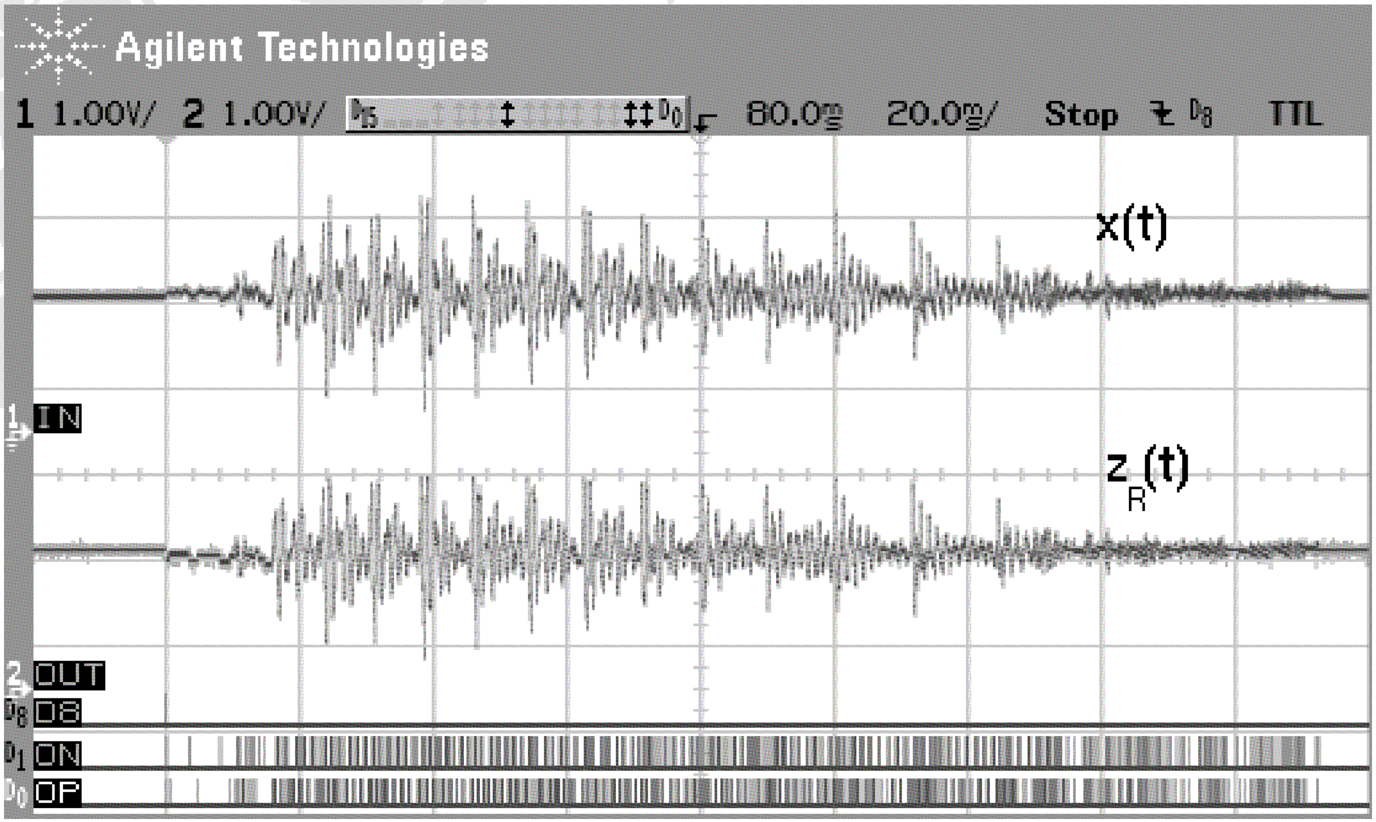
AER compatible signal

- Spike event coder signal *y(t)* – a digital signal
- Spikes may be transmitted over AER interface.
- Analogue signal reconstructed at receiving event coded building block.



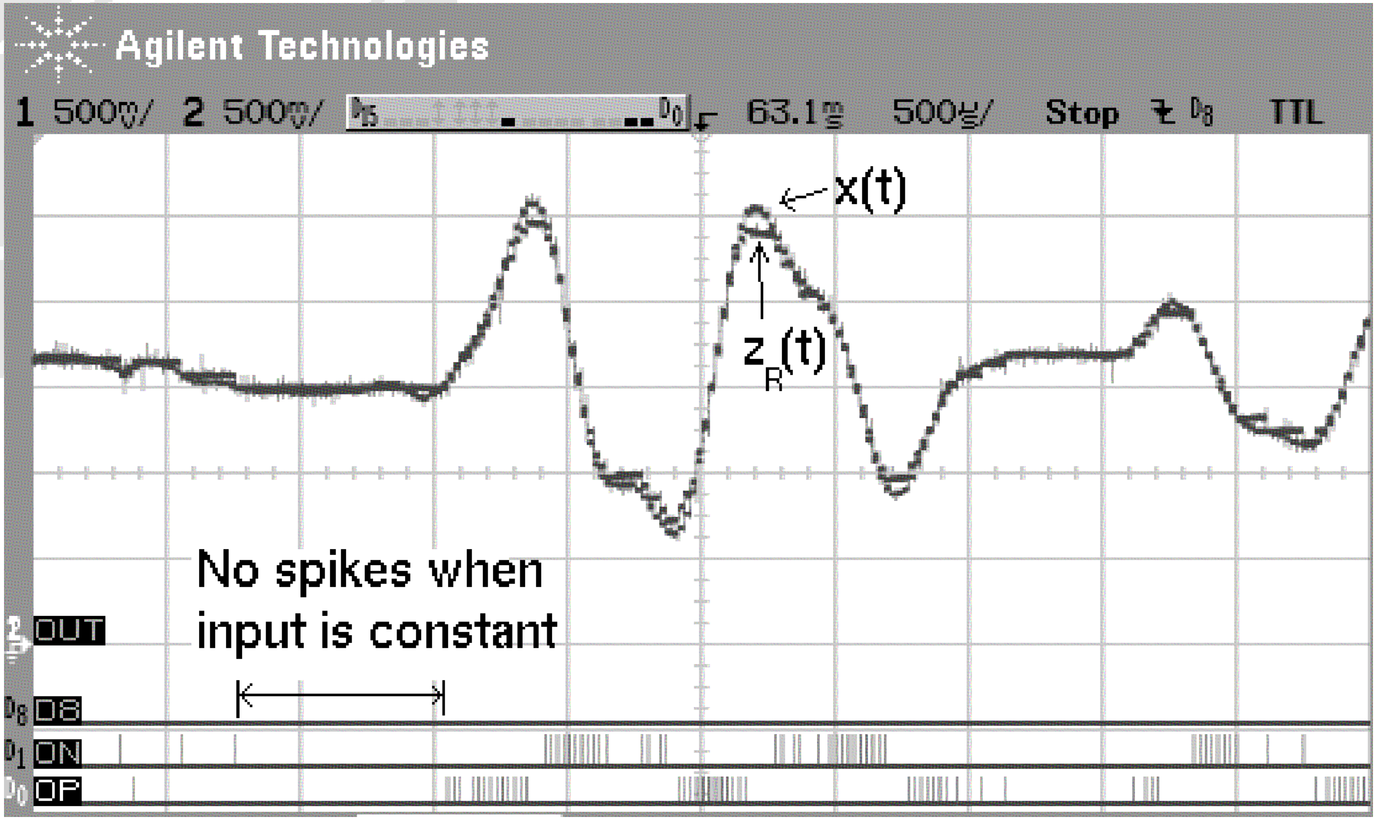
Speech signal reconstructed

- Original audio signal and reconstructed waveform



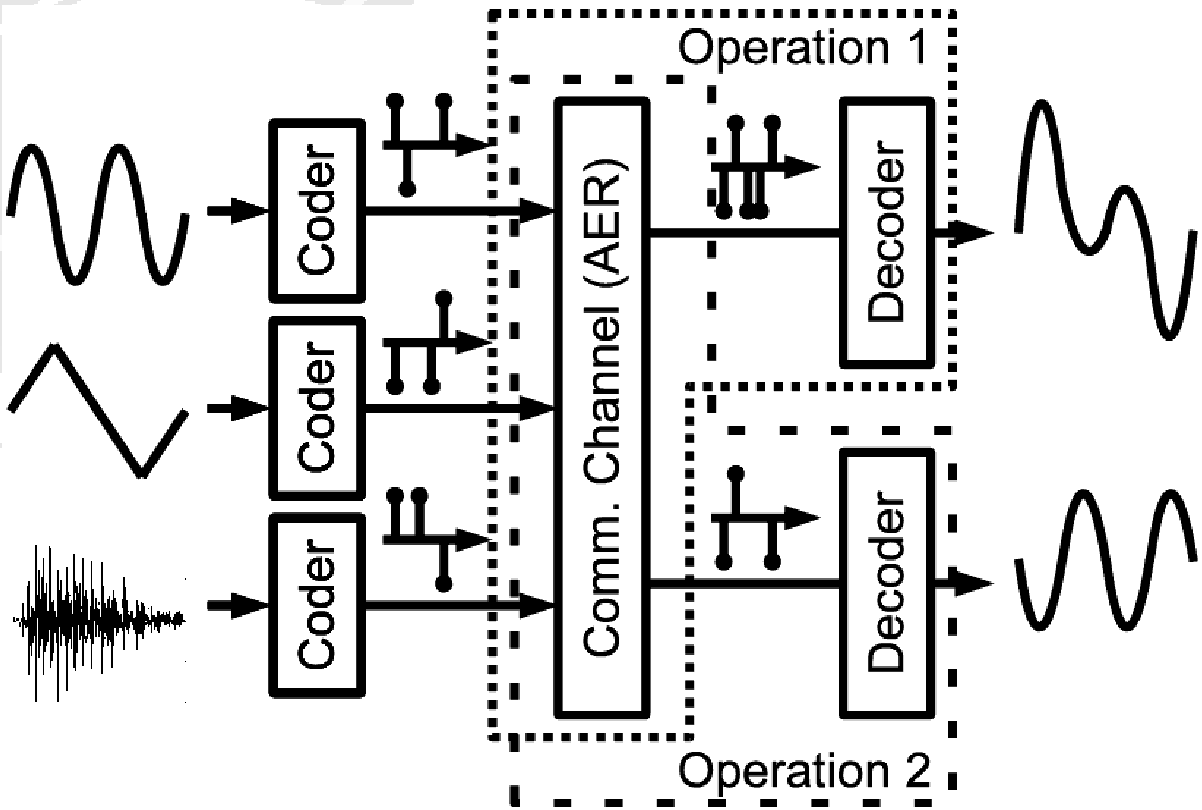
Speech signal – zoomed in

- Short segment of previous waveform



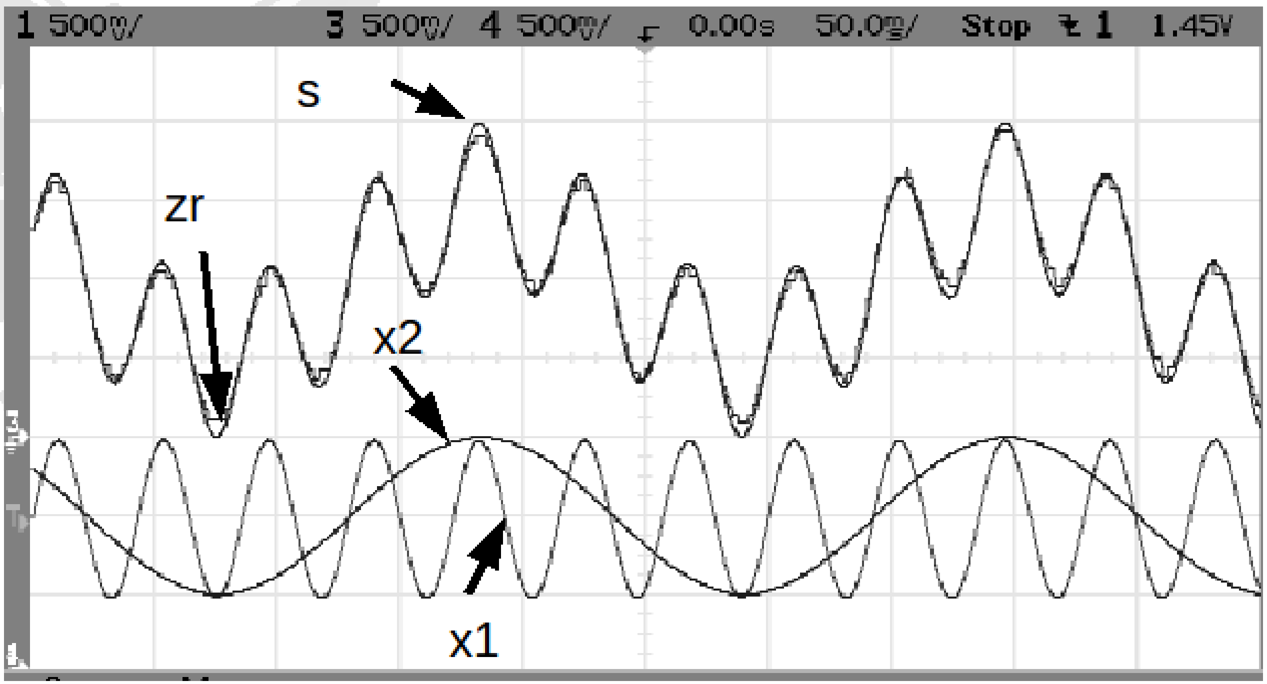
Simple arithmetic

- Shared AER bus allows scaling, adding of signals



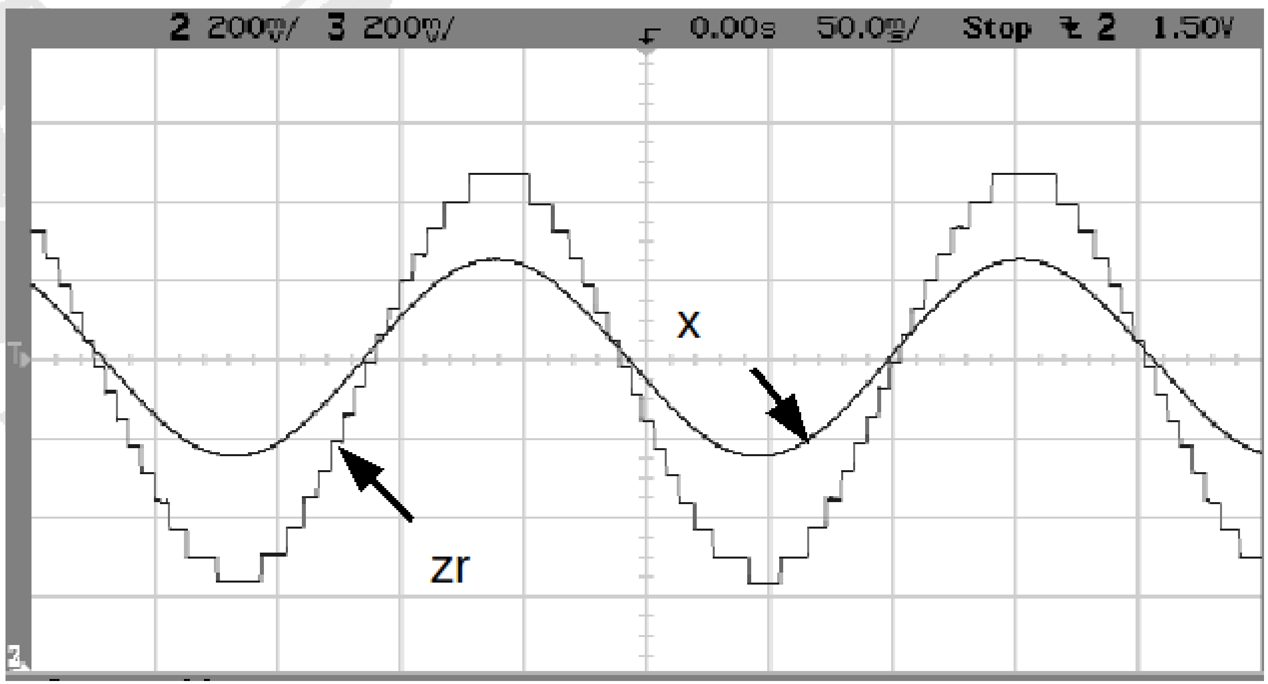
Signal addition

- Addition – sum of two sine waves



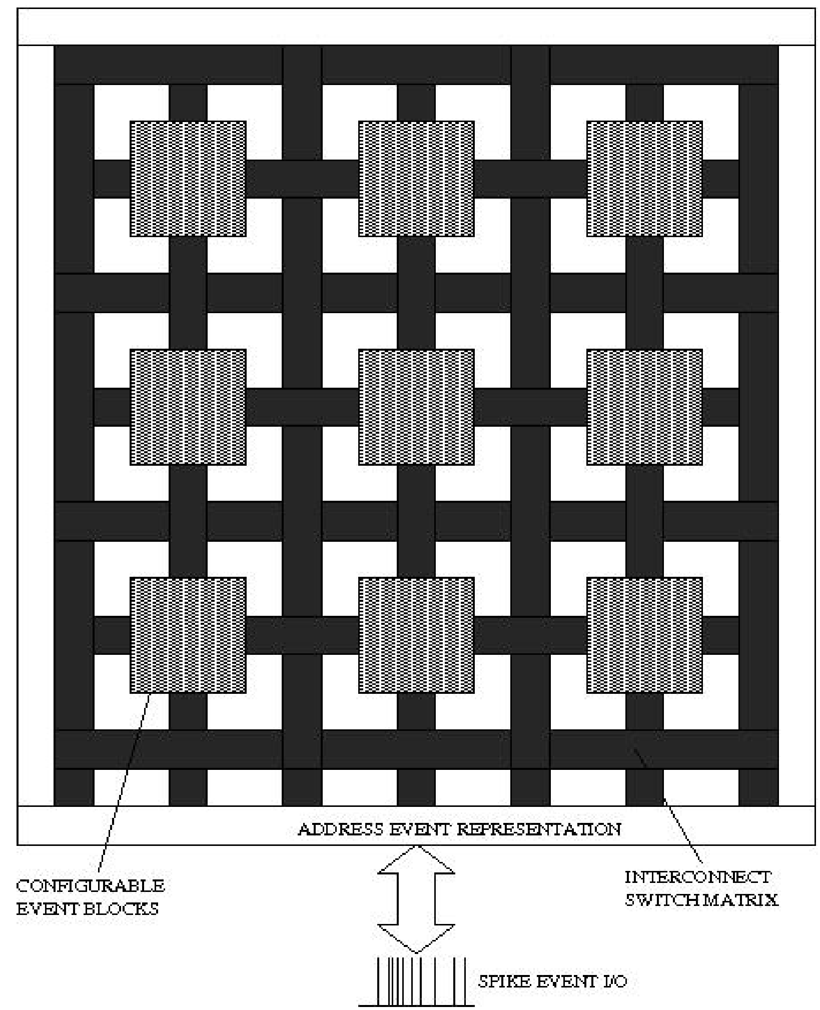
Signal amplification

- Amplification - ratio of integration scaling factors



Towards an event coded programmable analogue array

- CABs can perform all functionality required by a neuromorphic system e.g. our olfactory implementation.
- Analogue and spike based signals may be routed between CABs using event coding + AER.



Conclusions

- Novel programmable analogue array architectures developed at the University of Edinburgh
- ***Palmo***
  - A range of voltage and current mode circuits developed
  - Results from working analogue VLSI
- ***Event coding***
  - Basic units demonstrated in a range of applications
  - Work in progress on communication strategies

Key references

- *A Palmo Cell using Sampled-Data Log-Domain Integrators*, T. Brandtner et al, IEE Electronic Letters, Vol. 34, No. 8, pp 773-734, April 1998, ISSN 0013 5194
- *Palmo : Pulse Based Signal Processing for Programmable Analogue VLSI*, K. A. Papathanasiou et al, IEEE Trans. on Circuits and Systems, Vol. 49, No. 6, pp 379-389, June 2002
- *Programmable Analog VLSI Architecture Based Upon Event Coding*, T. J. Koickal, A. Hamilton and L. C. Gouveia, pp 554-559 NASA/ESA Conference on Adaptive Hardware and Systems, Edinburgh, UK, 5-8th August 2007, ISBN 0-7695-2866-X
- *An Asynchronous Spike Event Coding Scheme for Programmable Analog Arrays*, L. C. Gouveia, T. J. Koickal and A. Hamilton, IEEE International Symposium on Circuits and Systems (ISCAS), pp 1364-1367, 2008
- *A CMOS Implementation of a spike event coding scheme for analog arrays*, L. C. Gouveia, T. J. Koickal and A. Hamilton, IEEE ISCAS, pp 149-152, 2009
- *Computation in Communication: Spike Event Coding for Programmable Analogue Arrays*, L. C. Gouveia, T. J. Koickal and A. Hamilton, IEEE ISCAS, 2010