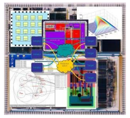
# Interfacing real-time spiking I/O with the SpiNNaker neuromimetic Architecture

Sergio Davies, Cameron Patterson, Francesco Galluppi, Alexander D. Rast, Steve B. Furber

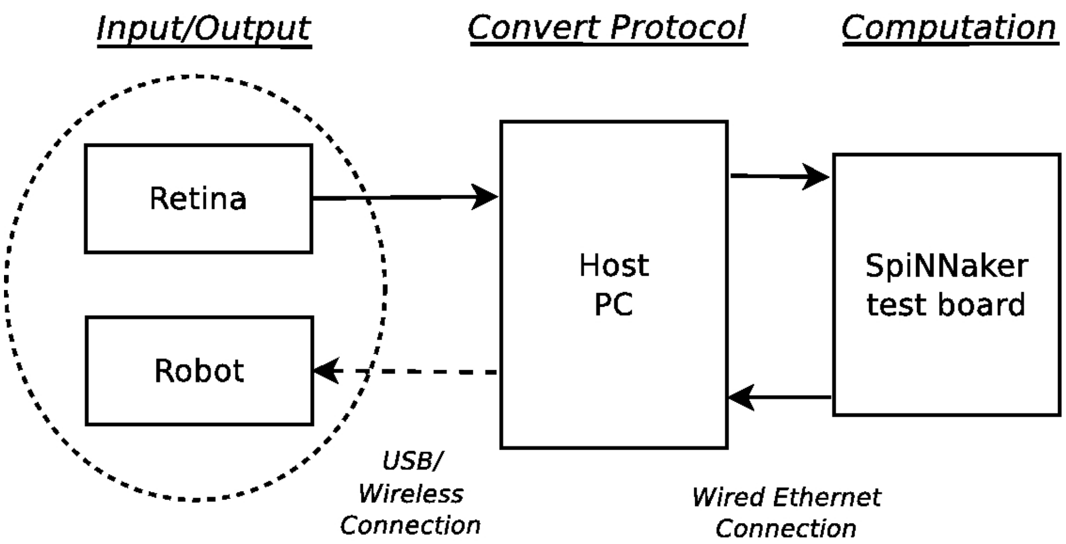
APT group
The University of Manchester

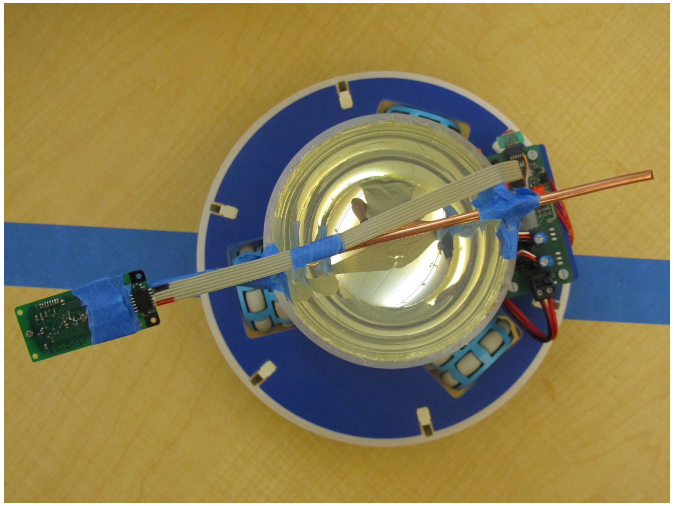


# Overview of the topics

- Test environment description
- Result shown on video
- Silicon retina description
- Robot actuators
- Neural network implemented

# Test environment description



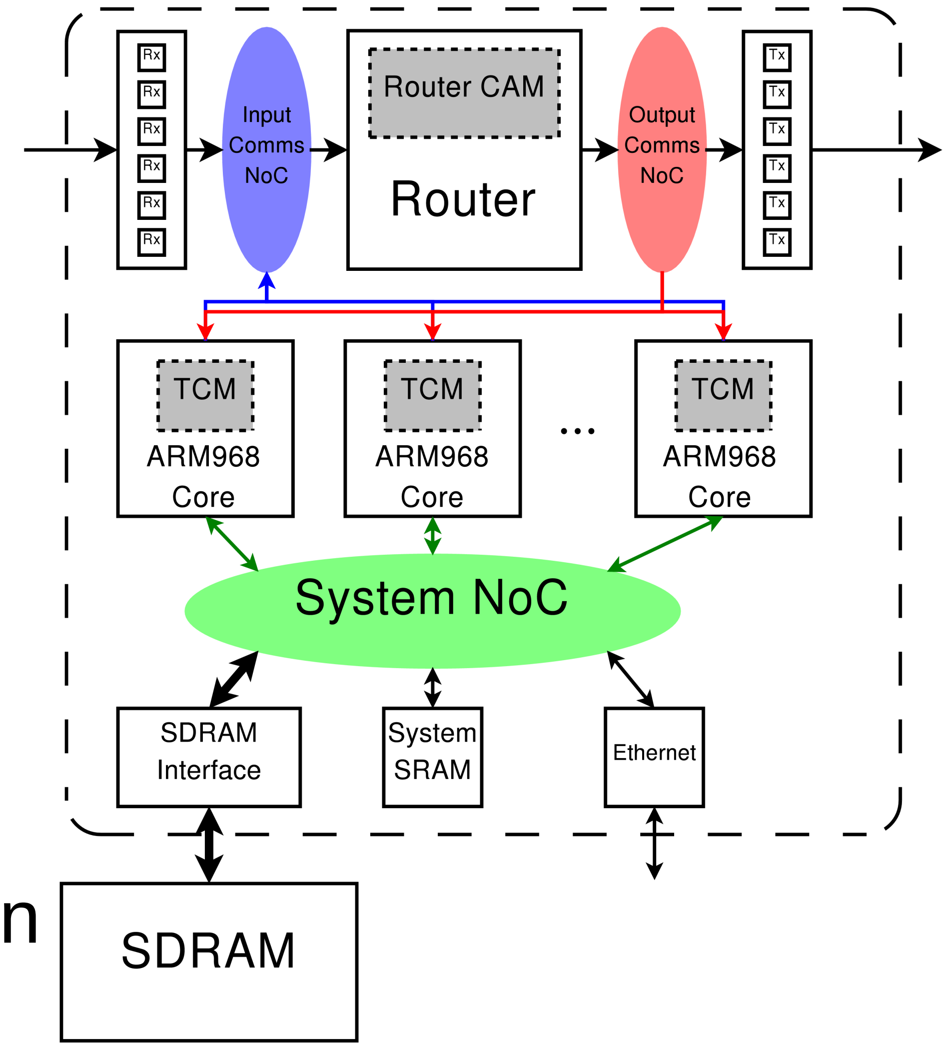



- Robot configuration:
  - Silicon retina (USB interface) on the top
  - Three wheels (wireless Ethernet interface) to move
- Spinnaker test board to simulate the neural network
- Host PC to convert interfaces and protocols

The University of Manchester

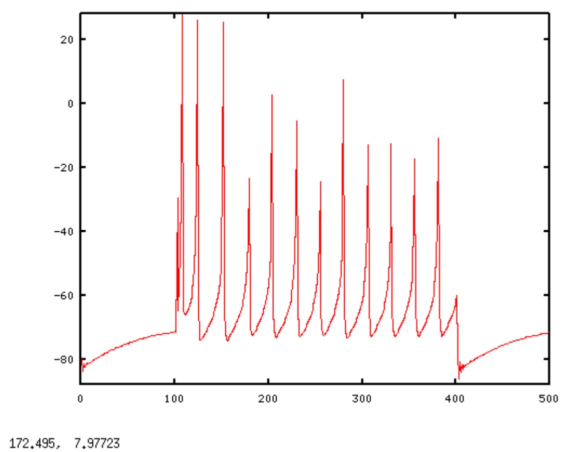
# Features of the SpiNNaker chip

- Native parallelism
- Event-driven processing
- Incoherent memory
- Incremental reconfiguration

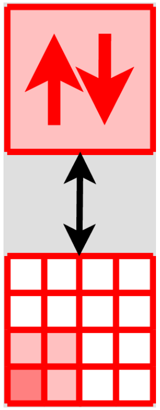



SpiNNaker a universal Spiking Neural Network architecture

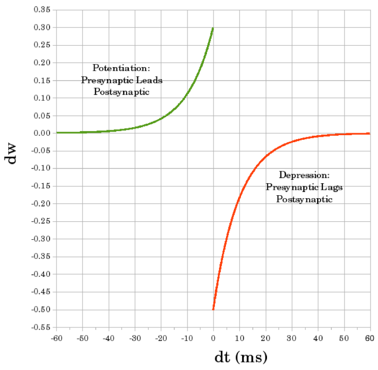
# Incoming spikes



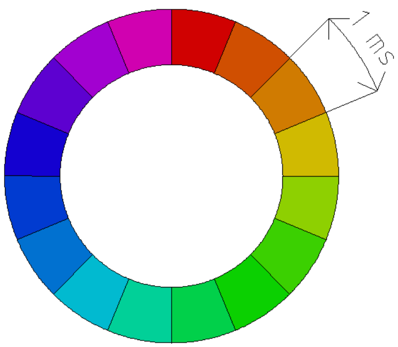

Spike incoming (interrupt received)



Retrieving synaptic weights

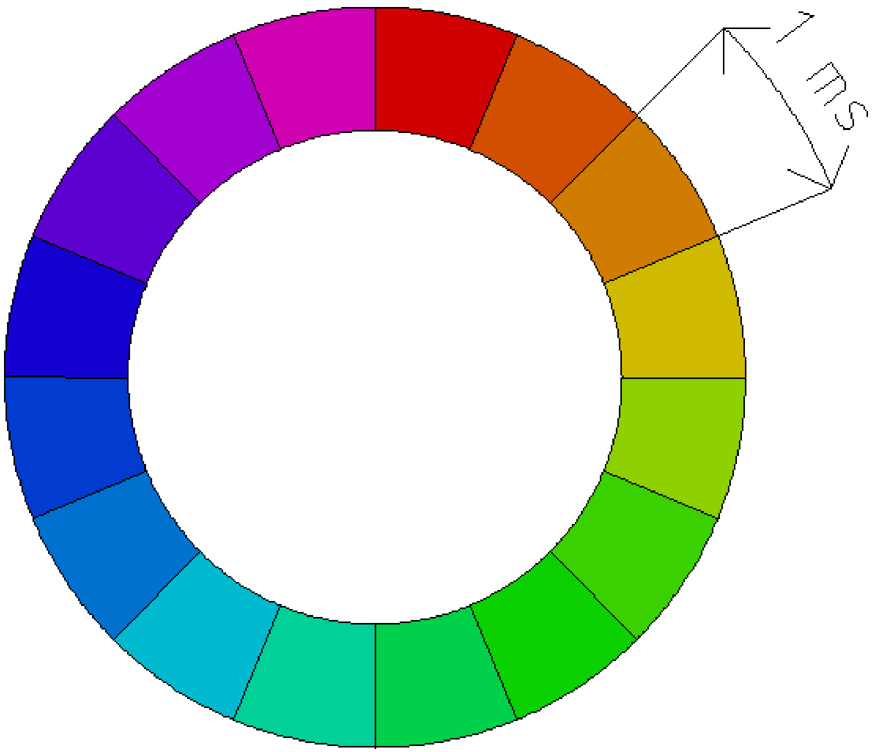



Synaptic plasticity (STDP)




Adding the new input in the delay buffer

# The delay buffer



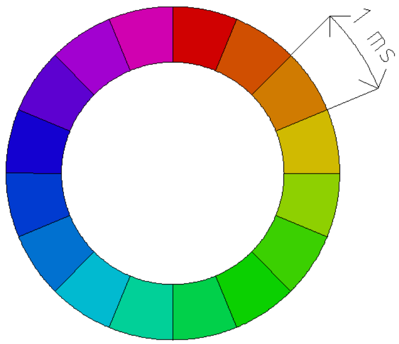

- 1 millisecond each slot (a.k.a. bin);
- 16 slots for a maximum delay of 16 millisecond;
- Incoming spikes adds synaptic weights in the correspondent slot;

MANCHESTER 1824

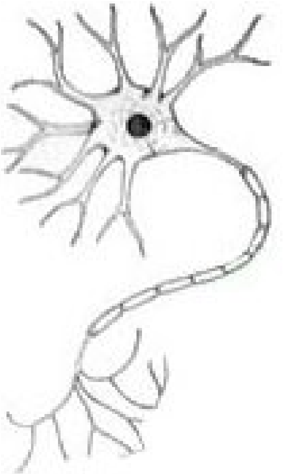
The University of Manchester

# Neural simulation

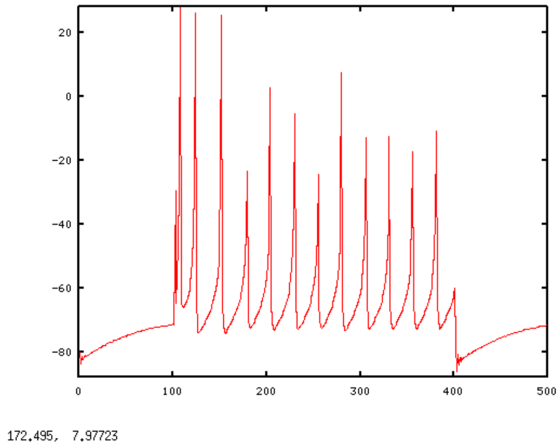
Timer interrupt




Neuron input



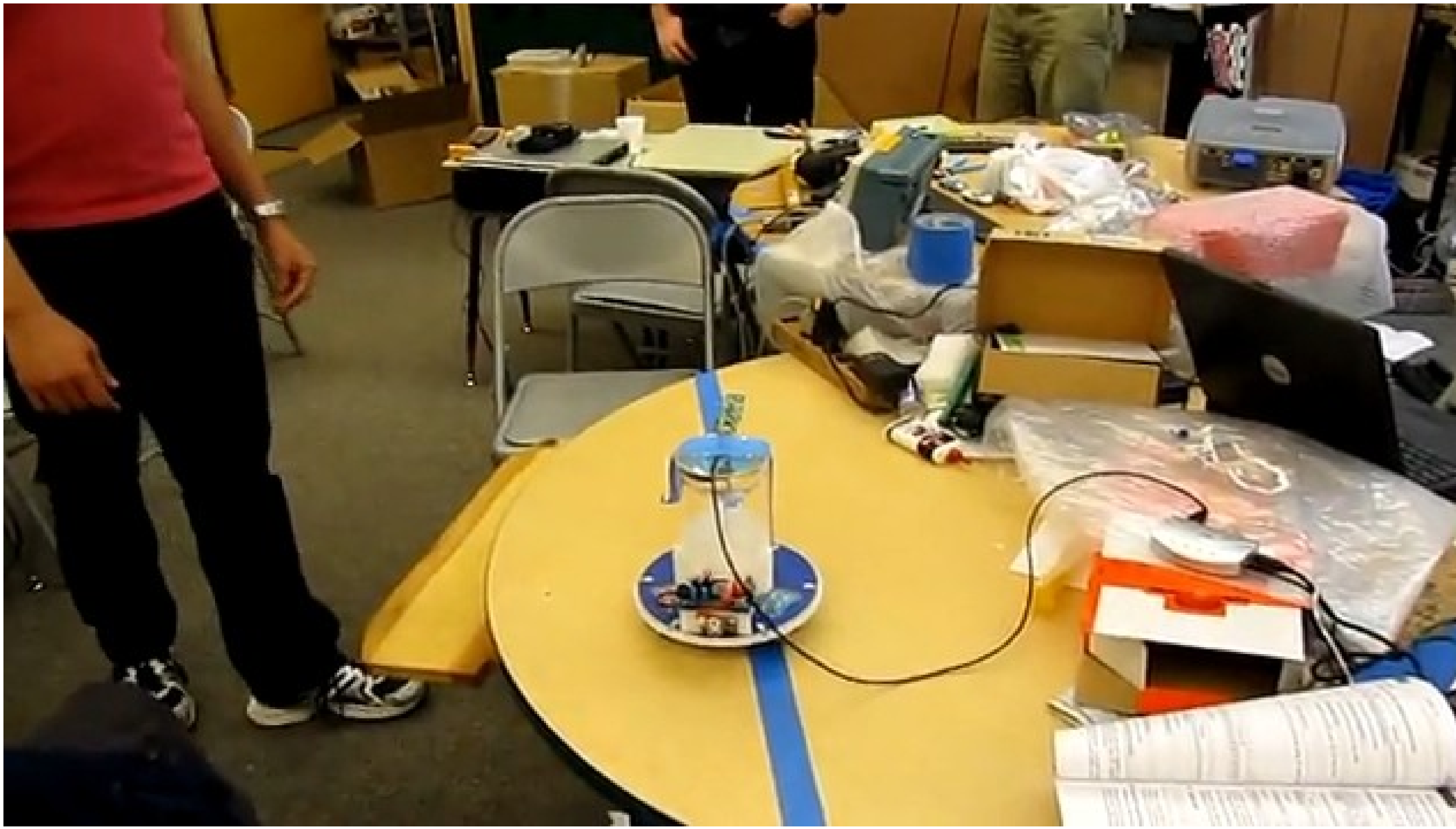
Differential equation computation




Spike emission

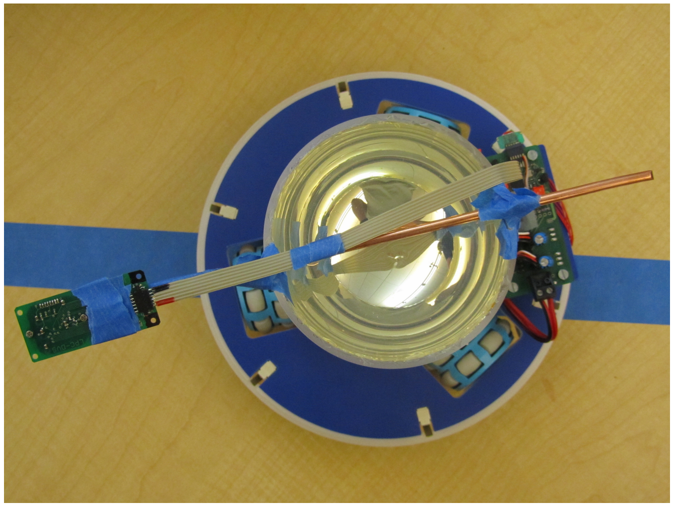
SpiNNaker a universal Spiking Neural Network architecture

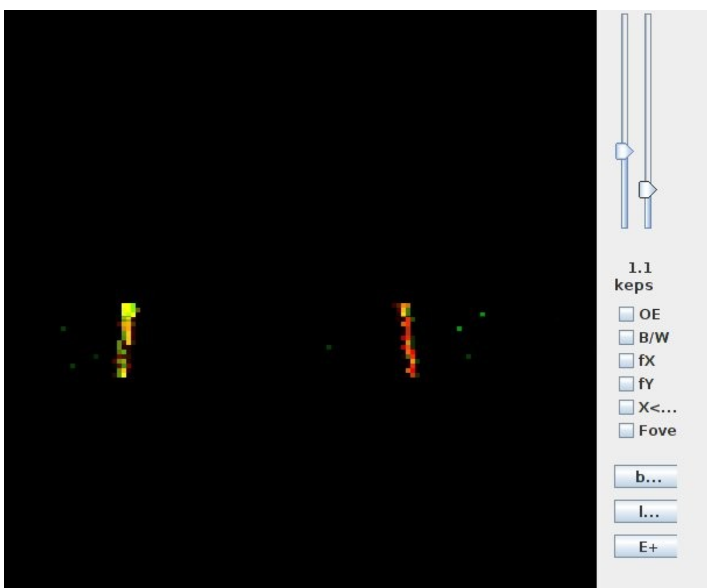
# Demonstration video



Video available on YouTube: “**SpiNNaker line following**”
http://www.youtube.com/watch?v=ZQ7FdQ_VJNg

# Silicon retina






The silicon retina has 128x128 elements reacting to light changes;

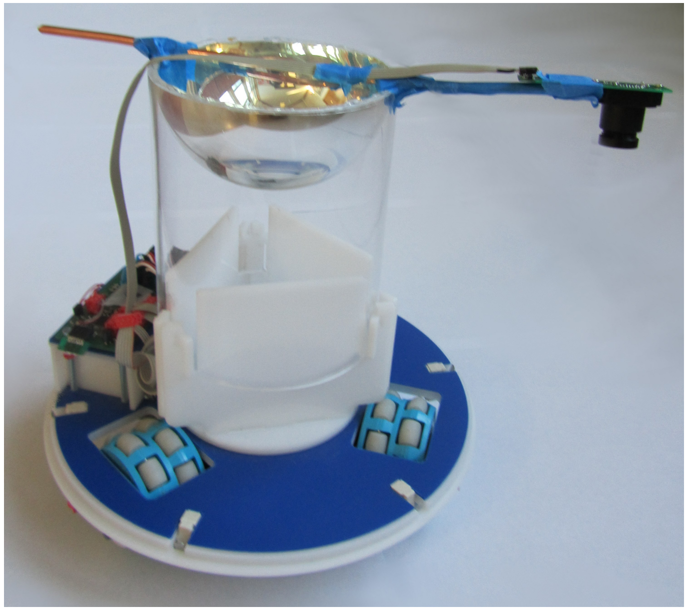
If the light intensity in a pixel varies more than a threshold the pixel emits an event (spike);

Too many pixels for the software, we took only the central part of the image (sub-sampling from 128x16 to 64x8);

MANCHESTER 1824

The University of Manchester

# Robot actuators

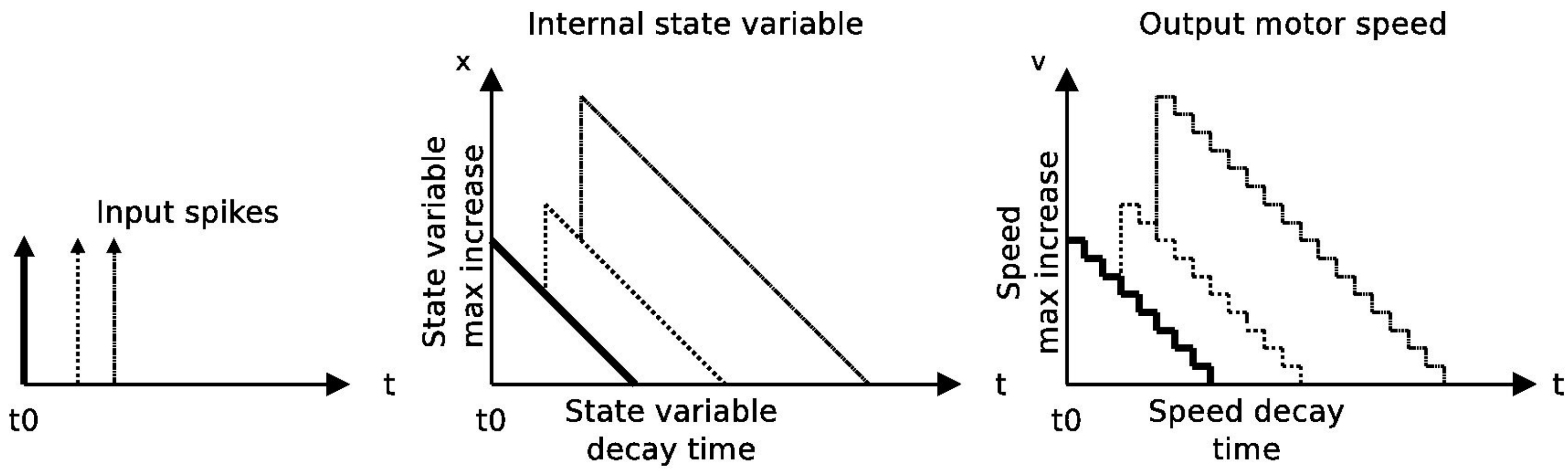


Actuators are three independent wheels, which are active in a specific direction and passive in the perpendicular direction.

The robot accepts command to set the speed in three directions: forward (Y axis), right (X axis) and rotation (on the robot central axis).

SpiNNaker a universal Spiking Neural Network architecture

# Robot actuators – speed coding



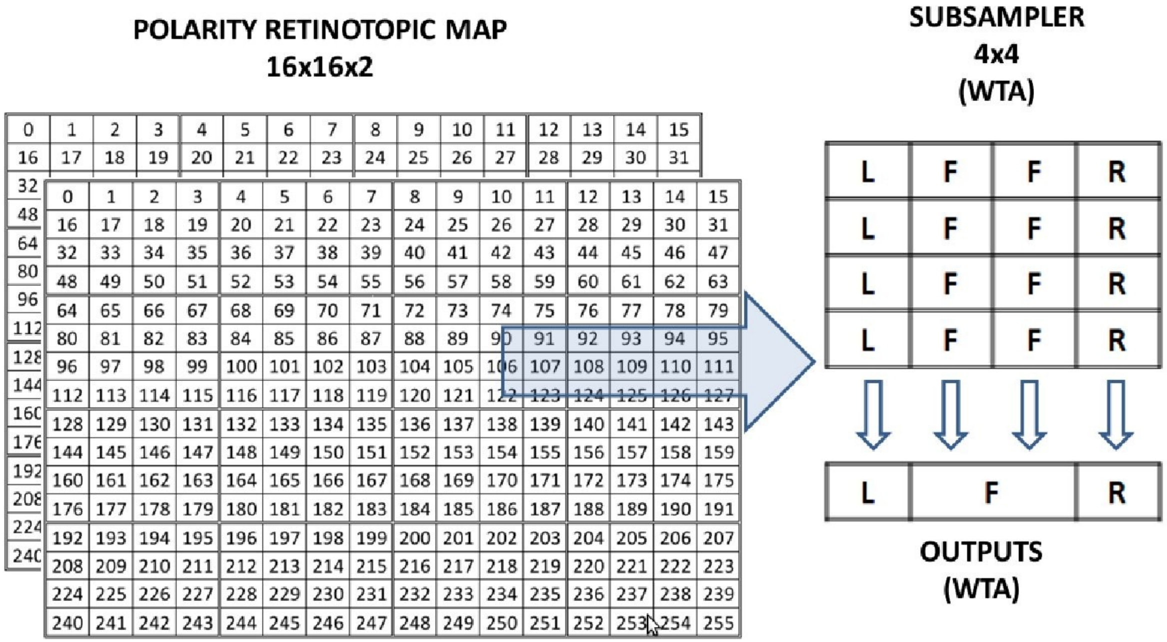

The speed of the robot in each direction is rate-coded: the more frequent the spikes are, the faster the robot goes in the correspondent direction.

This is achieved through a state variable in the conversion between spikes and speed.

MANCHESTER 1824

The University of Manchester

# Neural network implemented




The retina events are routed to 512 neurons (256 for each polarity) through a sub-sampling.

The computation layer sub-samples the input to a 4x4 neuron matrix.
The output neurons control the forward, left and right directions.

SpiNNaker
a universal Spiking Neural Network architecture

# Conclusions

- The robot was successfully able to follow a line;
- SpiNNaker chip was able to interface in real-time with spiking I/O in an autonomous closed loop system;
- No learning was involved in this process;
- This is the first step towards the goal of an autonomous robotic system using bio-inspired spike-based signalling;
- Future work will involve also synaptic plasticity to enable the robot to understand the pattern of input, to follow and search autonomously.

MANCHESTER 1824

The University of Manchester

# THANK YOU!!!


SpiNNaker
a universal
Spiking Neural Network
architecture

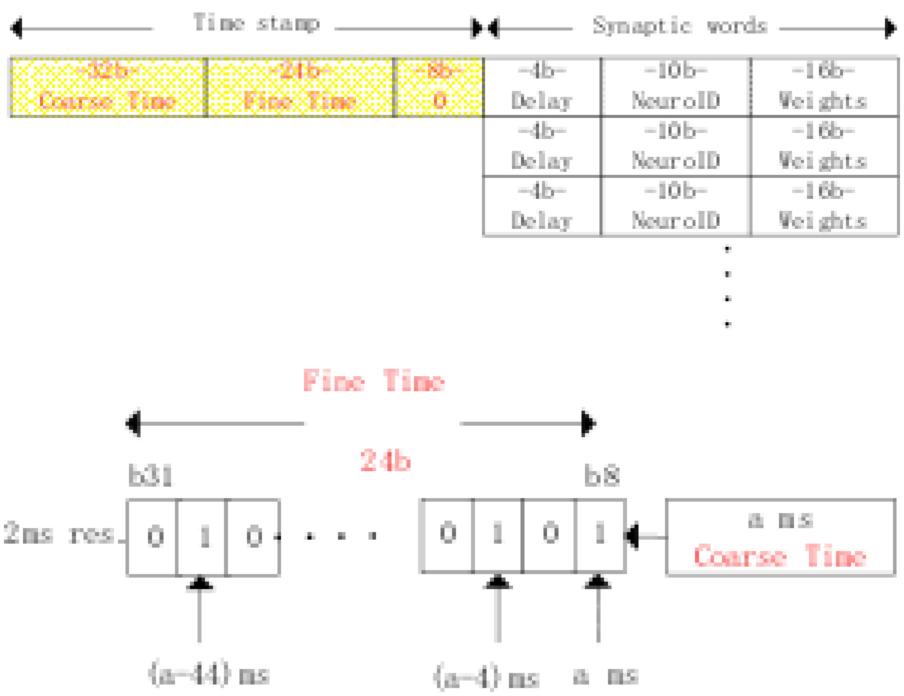
# Back-up slides

# Implementation

## Representation of spike timestamp

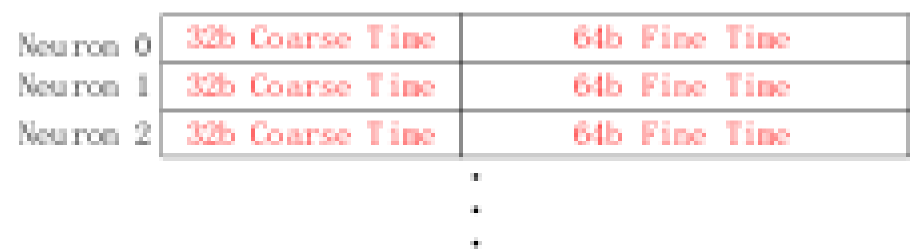
### Pre-synaptic timestamp

Needed only when a pre-synaptic spike arrives. Stored as header of the synaptic weight block



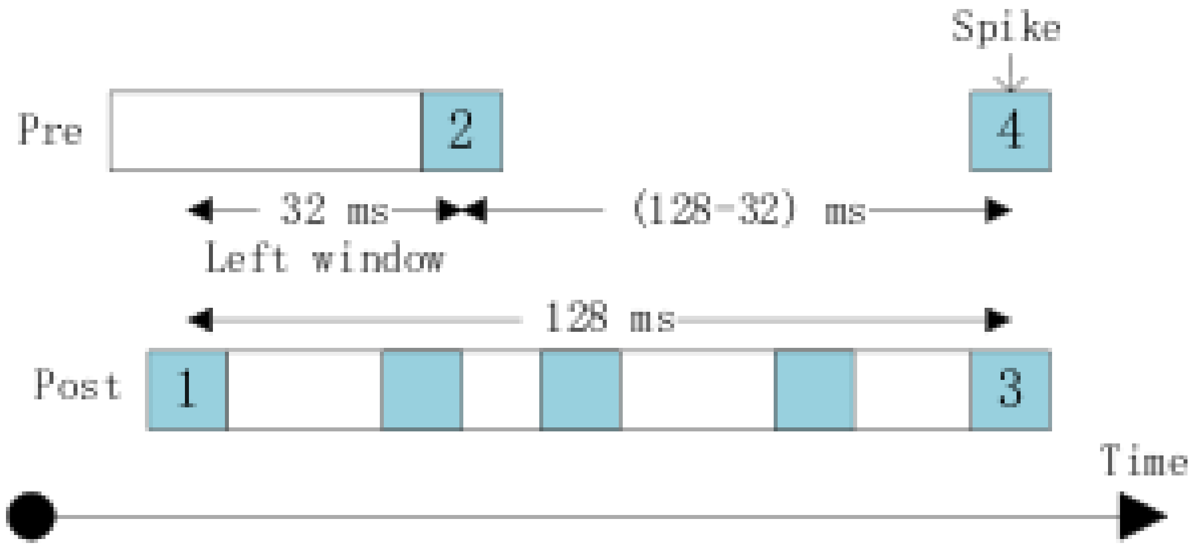

### Post-synaptic timestamp

Needed at all time. Stored in processor's local memory



# Implementation

## Length of timing records




The STDP is triggered when an incoming spike pushes an old input record into the carry bit
However, if the input arrives at very low rate the output generated pushes forward the previous records and the history will be lost.

# Neural network chip architectures

## NEUROPROCESSOR

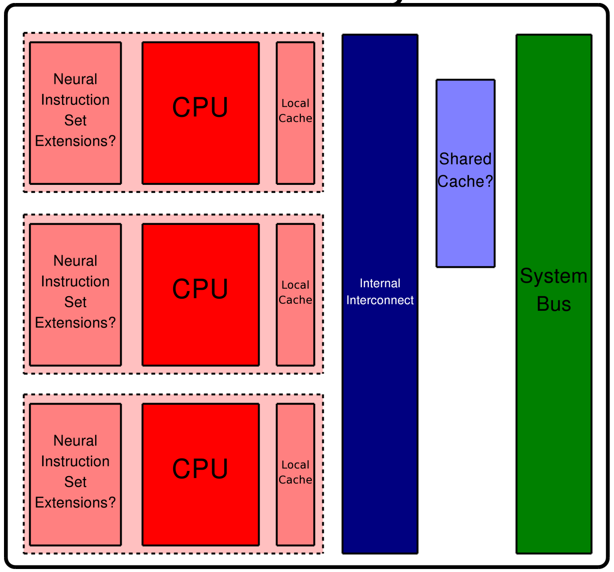
- Domain-specific multiprocessors
- High programmability
- Limited biological fidelity
- Minimal exploitation of intrinsic neurodynamics

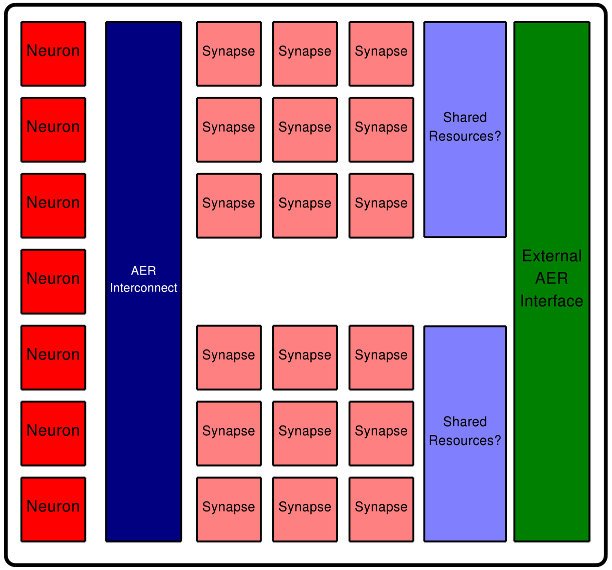
## NEUROMORPHIC

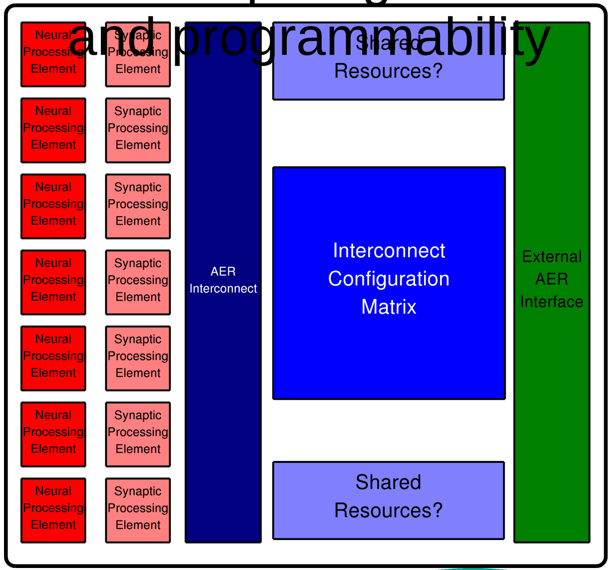
- Application-specific neuroprocessors
- Limited model support
- High biological fidelity
- Minimum exploitation of configuration

## NEUROMIMETIC

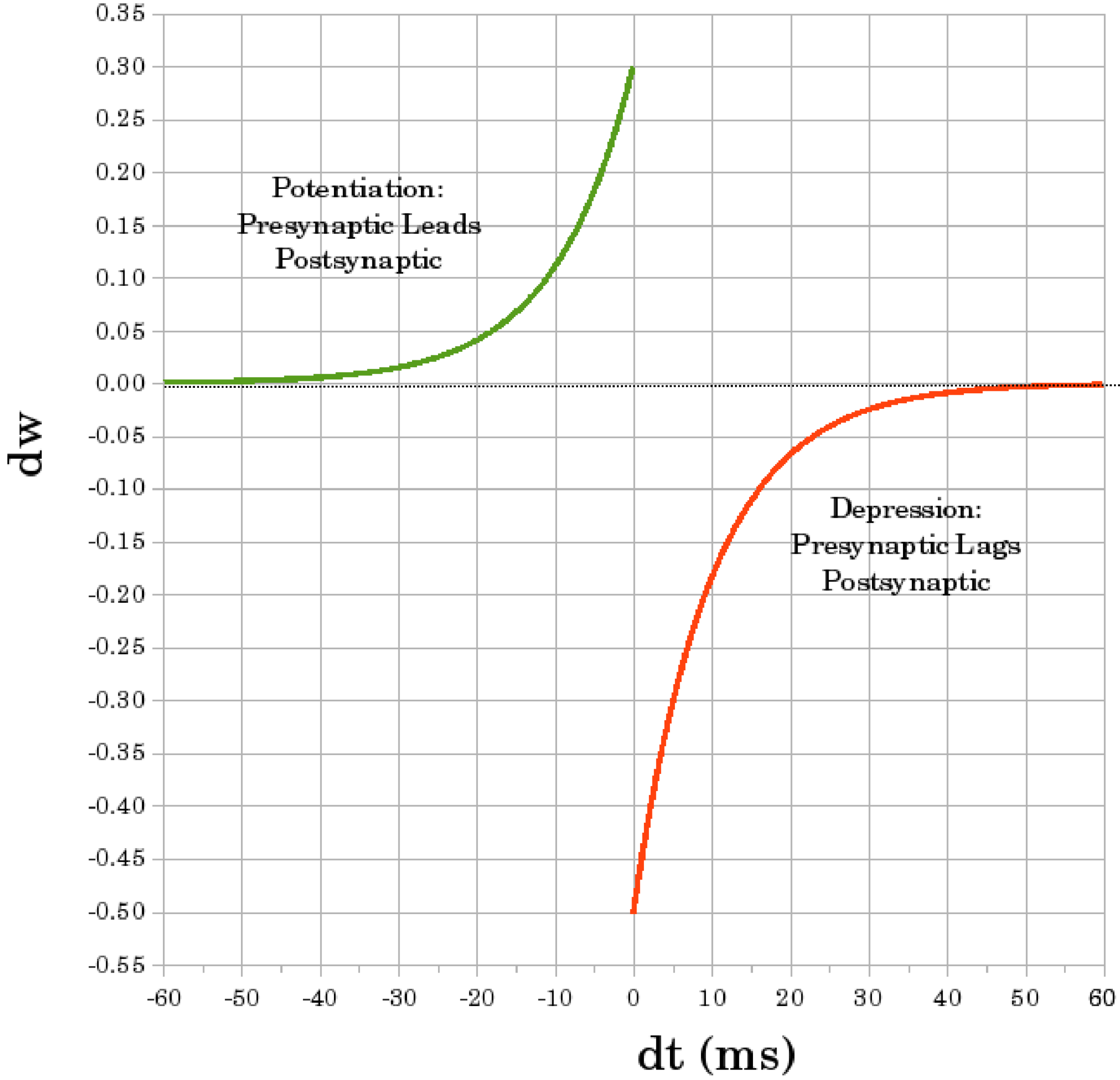
- Universal neuromorphic chip
- Dynamic configurability
- Tunable biological fidelity
- Balance neurocomputing and programmability

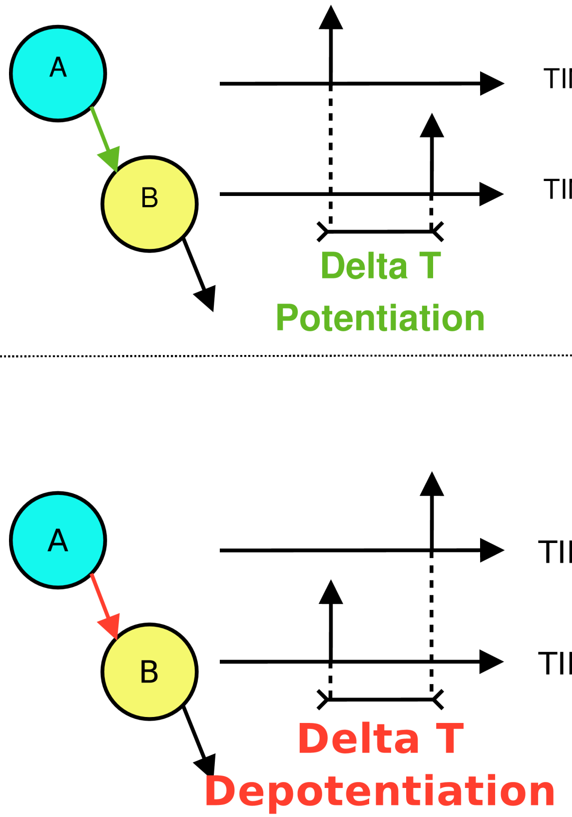


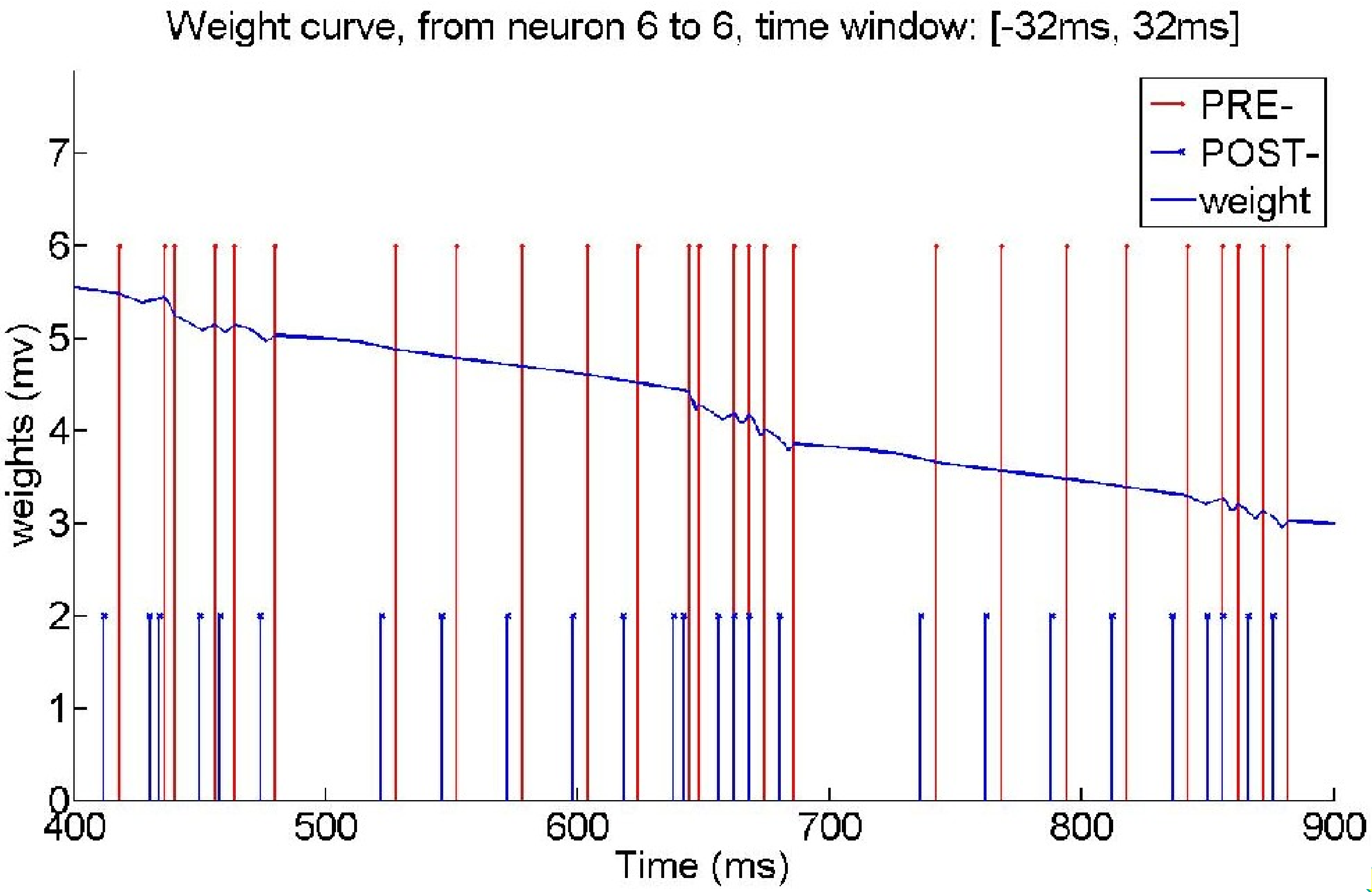



# Spike Timing Dependent Plasticity



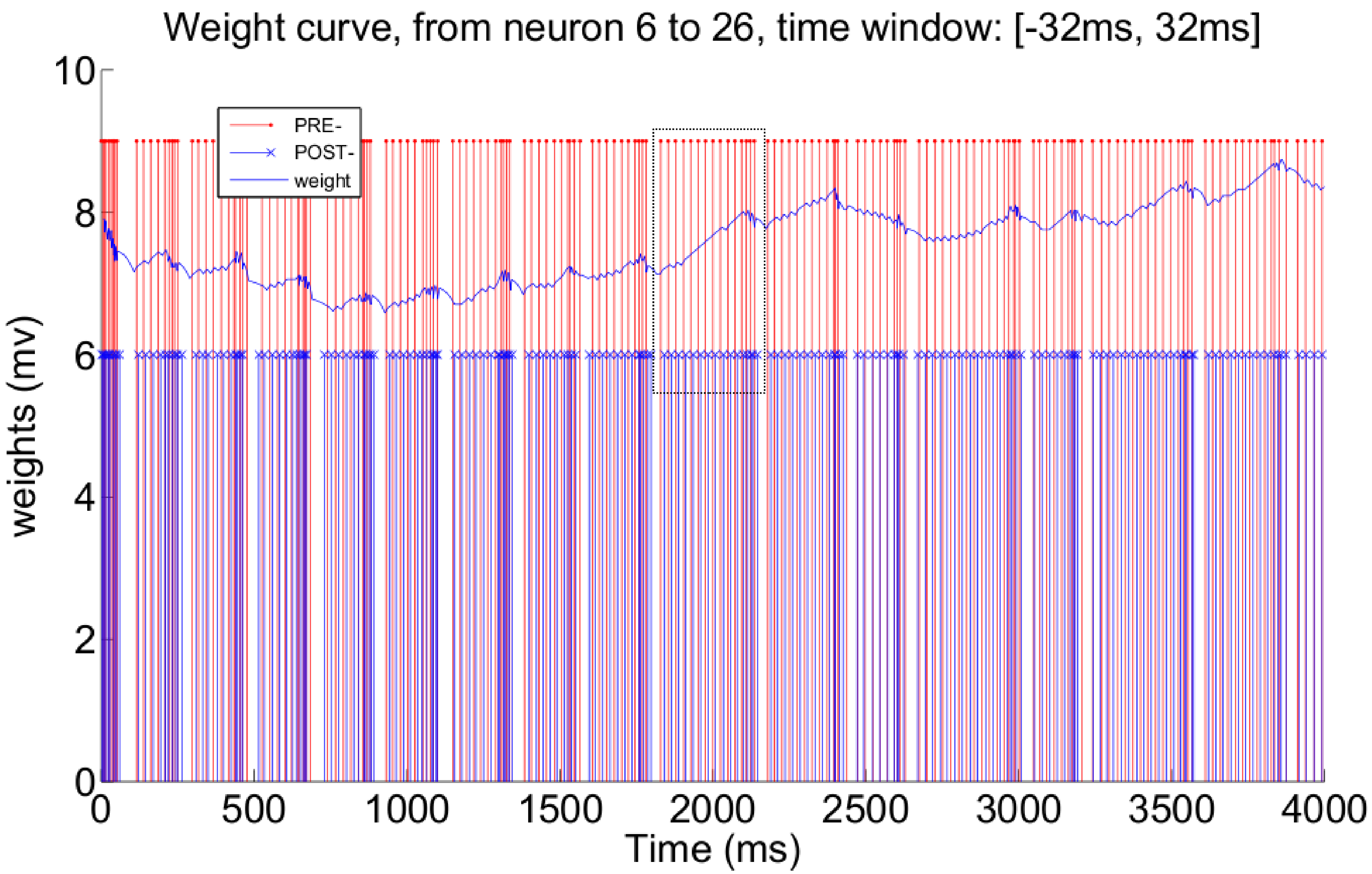



# LTD - example

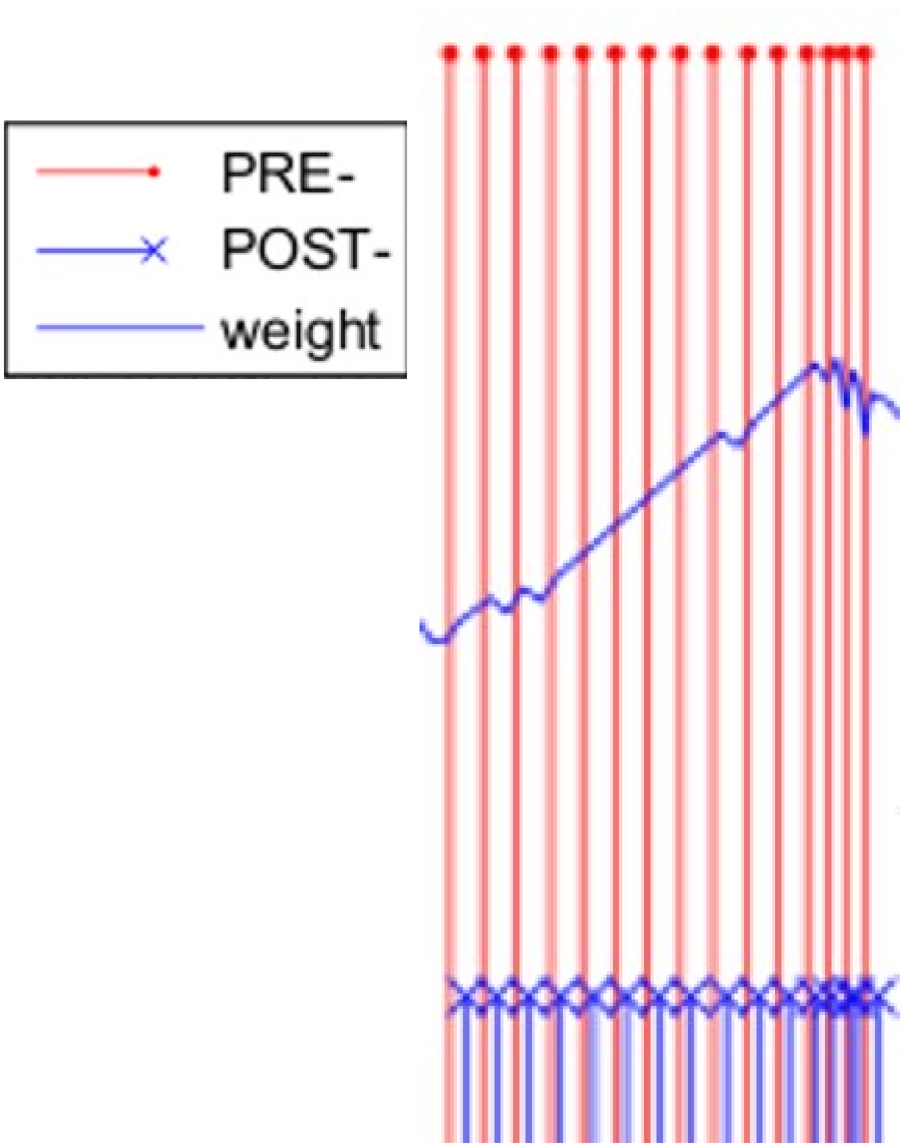

Weight curve, from neuron 6 to 6, time window: [-32ms, 32ms]
PRE-
POST-
weight
weights (mv)
Time (ms)
0
1
2
3
4
5
6
7
400
500
600
700
800
900

The University of Manchester

# LTP - example



# LTP – example – details

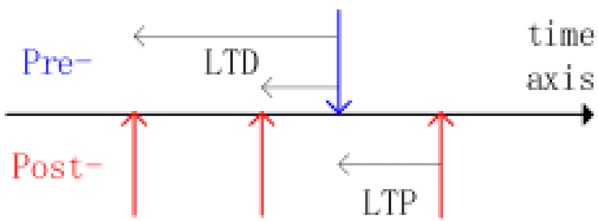



- Each sequence of pre-post synaptic spike generates an increase in the synaptic weight.
- When the pre-synaptic and the post-synaptic spikes are too close, the weight starts to oscillate rapidly

# Implementation

## Triggering the STDP algorithm

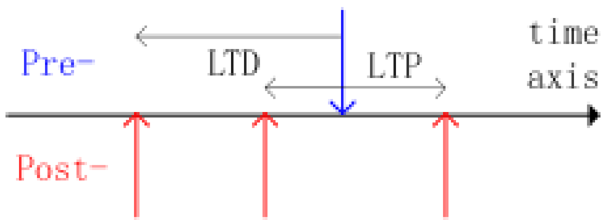
### The usual way:




STDP is triggered on:

- Pre-synaptic spike arrival (LTD)
- Post-synaptic spike emission (LTP)

### The SpiNNaker way:




- STDP is triggered only on pre-synaptic spike arrival (LTD and LTP)
- Weights are available only at pre-synaptic spike arrival.
- Since LTP needs future information, the algorithm needs to be deferred until the time window is filled

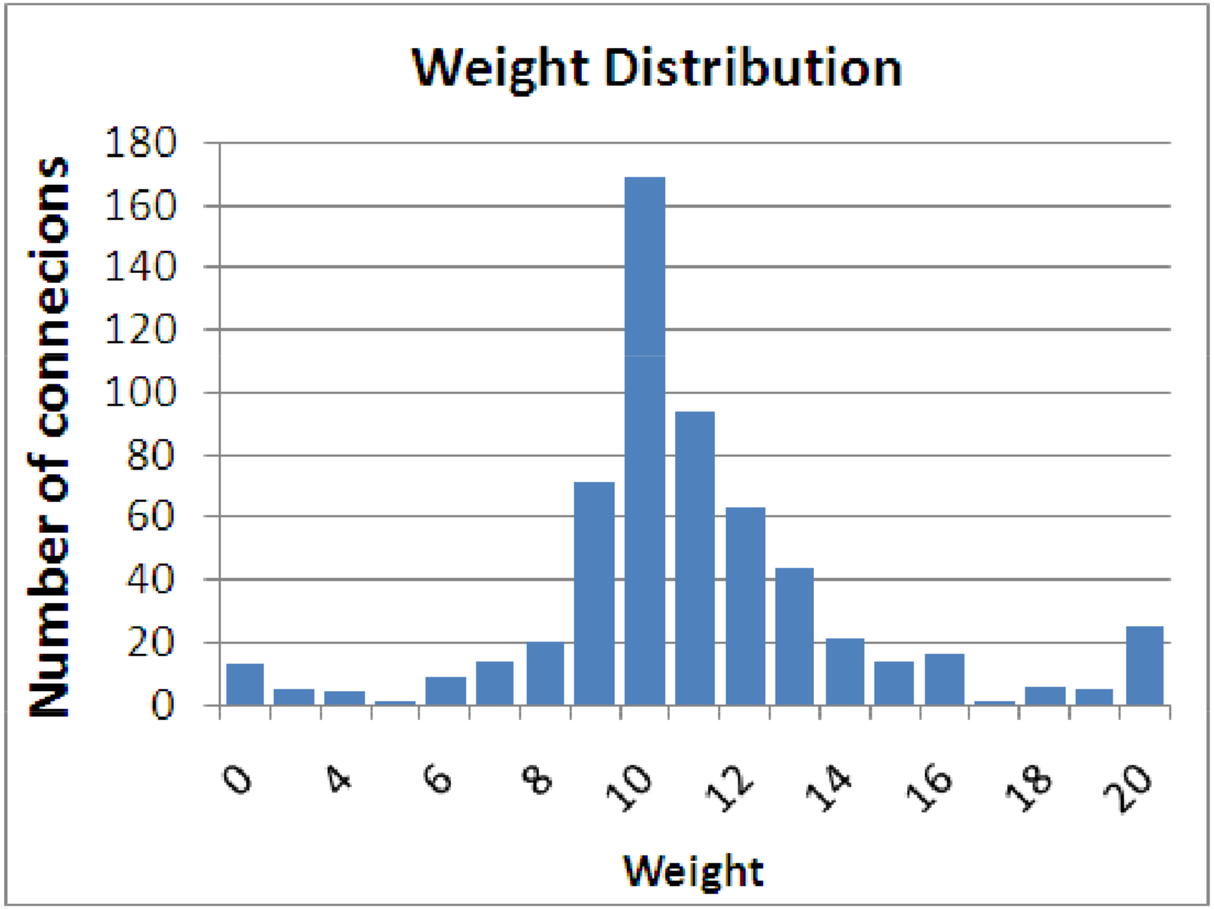
The University of Manchester

MANCHESTER 1824

# Network parameters

- Number of neurons: 76 – 60 excitatory, 16 inhibitory
- Type of neuron: Izhikevich model – exc: TS, inh: FS
- Simulation time: 30 seconds
- Starting weights: exc weights set at 10, inh set at -8;

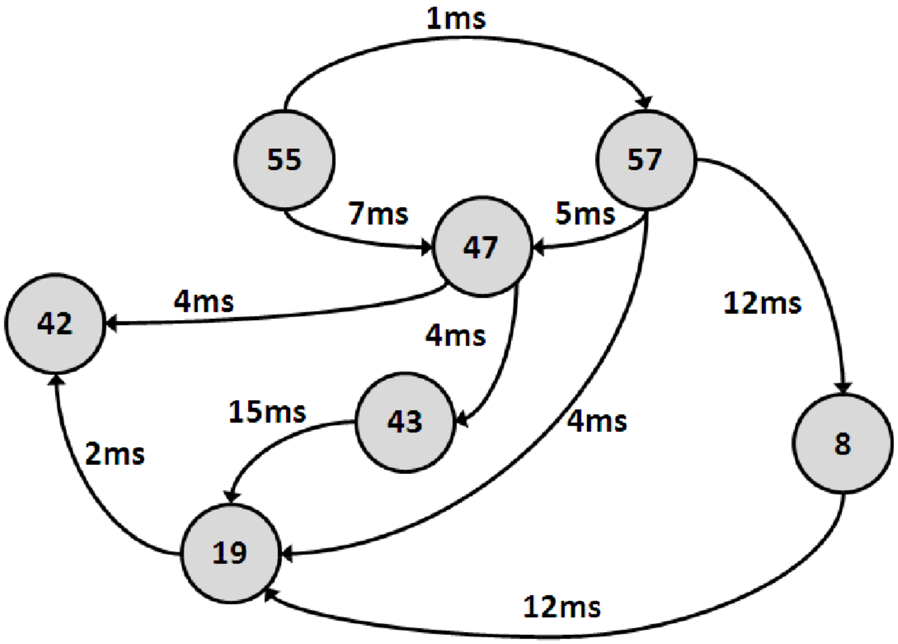
(inspired by Izhikevich, "Polichronization")




SpiNNaker a universal Spiking Neural Network architecture

# Results

In a simulation we run there were a group of 7 neurons which were strongly interconnected at the end of the simulation




Three circuits with converging delays:

- 57 → 8 → 19 and
  57 → 47 → 43 → 19
- 55 → 57 → 47 and
  55 → 47
- 55 → 57 → 19 → 42 and
  55 → 47 → 42

These connections are systematically reinforced due to the converging delays which makes the neurons fire in a pattern.

# Future work

- Rate – based plasticity
- Propagation delay plasticity
- Homoeostasis
- Rewiring