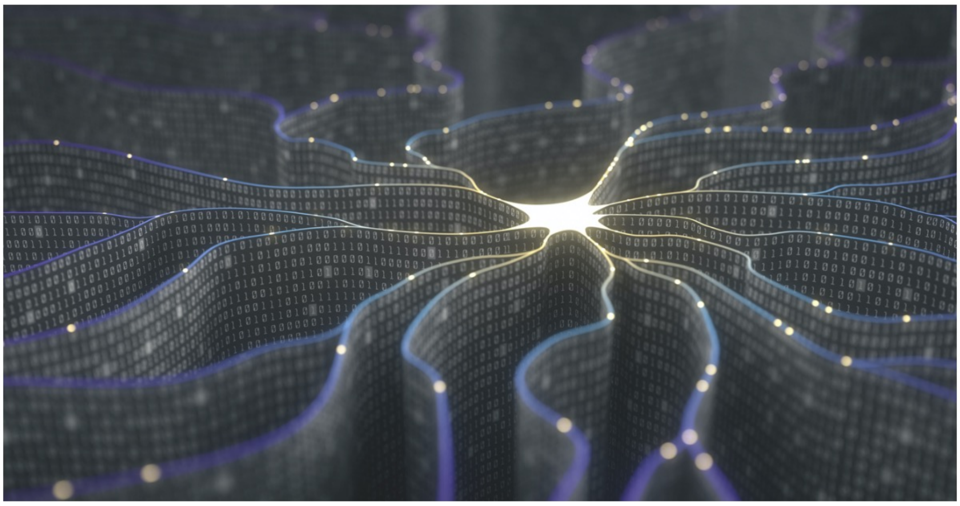
Liège University

Neuromorphic Signal Processing

Exploring Symmetry Breaking in Symmetric Neuron Networks



BOXUS Florent s201766
HALA Pierre s200712

Teachers : A. Franci
L.Mendolia

Academic year : 2023-2024

Table des matières

0.1 2 neurons case

The main utility of Julia and its bifurcation kit in this project is to be able to determine the value of the different parameters. To achieve this goal, the bifurcation diagram of the output current as a function of I_{gain} will be plotted. For that reason, the equations need to be adapted. Also, the interactions between the two neurons will be added. Two cases will be considered. In the first place, the two neurons will be mutually inhibiting each other. Afterwards, each neuron will excite the other one.

0.1.1 2 neurons - Mutual inhibition

The representation of the mutual inhibition between the 2 neurons can be seen in figure **??**. The idea is that the first neuron receives as input its constant input current (I_{in1}) to which we subtract the output of the second neuron (I_{out2}). The same idea is used for the input of the second neuron. So, the real input currents received by the two neurons are :

$$I'_{in1} = I_{in1} - I_{out2}$$
$$I'_{in2} = I_{in2} - I_{out1}$$

The introduction of the dependency of the input of a neuron by the output of the other neuron is included in I_{cmp} since I_{cmp} depends on V_{in}.

Also, since we will generate the bifurcation diagram of I_{out} as a function of I_{gain}, the I_{out} function should also depend on V_{gain} and will thus be defined by :

$$I_{out}(V_{out}, V_{gain}) = I_0 \frac{e^{\kappa V_{out}/U_T}}{1 + e^{\kappa (V_{out} - V_{gain})/U_T}}$$

First try
In the first attempt of plotting the bifurcation diagram of the output current of the first neurons (I_{out1}) as a function of the gain current (I_{gain}), the input currents were fixed at $300\,nA$ ($I_{in1} = I_{in2} = 300\,nA$) with $I_{thr} = 100\,nA$ and $I_{lin} = 300\,nA$. The bifurcation diagram obtained is :

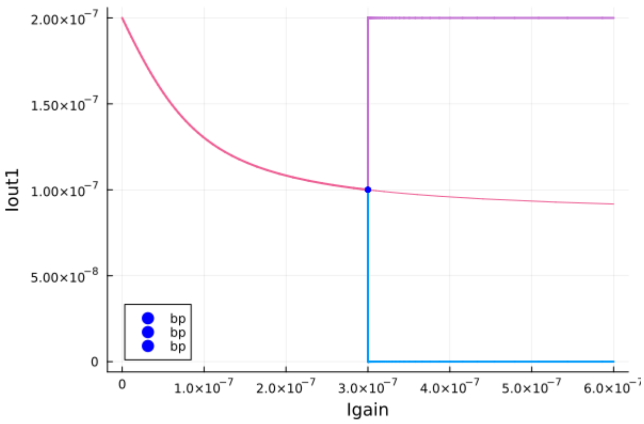


FIGURE 1 – First try of the bifurcation diagram of I_{out1} as a function of I_{gain}

It can be observed the presence of a bifurcation point at $I_{out1} = 1 \cdot 10^{-7}\,A$ and $I_{gain} = 300\,nA$. The bifurcation is really step because I_{out} passes from $100\,nA$ when $I_{gain} = 300 - \Delta\,nA$ toward either $I_{out} = 0\,A$ or $I_{out} = 200\,nA$ for the stable branched when $I_{gain} = 300 + \Delta\,nA$ with $\Delta \to 0$. The system is really sensible to a really small increase of I_{gain} after reaching $I_{gain} = 300\,nA$. This bifurcation corresponds to a supercritical pitchfork bifurcation since we pass from a single equilibrium from which emerge symmetrically 2 stables equilibrium and a unstable equilibrium in the center.

It must be remarked that if the bifurcation diagram of I_{out2} as a function of I_{gain} was done, the same result would be obtained since the system is totally symmetric.

The main problem with the bifurcation diagram obtained was the fact that I_{out1} was starting from $200\,nA$ instead of zero when $I_{gain} \to 0$. Another problem was the fact that after the bifurcation, the unstable middle branch was decreasing. Those behaviours were absolutely not the ones expected and the ones that should be observed. For that reason, we needed to find alternative.

Second try

For the second attempt, we have found a way to make a bifurcation in which I_{out} was starting at zero for I_{gain} as expected. To be able to observe that behaviour, we needed to fix the input currents of the neurons to the gain current. This means that $I_{in1} = I_{in2} = I_{gain}$. As a consequence, those currents will evolve as I_{gain}. Considering that adaptation and the inhibition of the output current of the other neuron, the real input receives by the neurons are :

$$I'_{in1} = I_{gain} - I_{out2}$$
$$I'_{in2} = I_{gain} - I_{out1}$$

The corresponding bifurcation diagram of I_{out1} as a function of I_{gain} is given by the following figure.

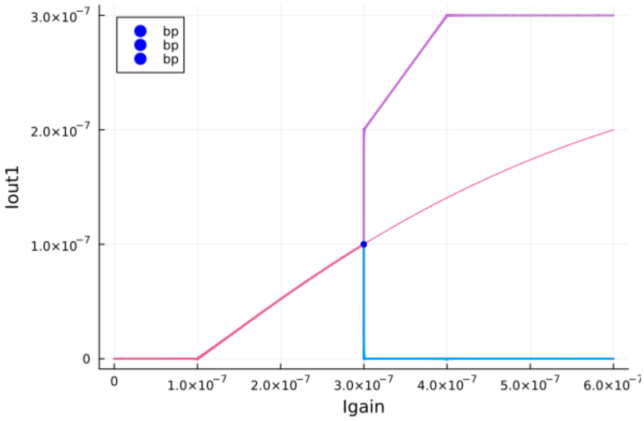


FIGURE 2 – First try of the bifurcation diagram of I_{out1} as a function of I_{gain}

As wanted, I_{out} begins from zero and starts to increase when $I_{gain} = I_{thr} = 100\,nA$. The bifurcation point is at $I_{out1} = 100\,nA$ and $I_{gain} = 300\,nA$ as for the first try. This bifurcation correspond also to a pitchfork bifurcation but this time it is asymmetric. The asymmetry can be seen through the fact that the lower and upper stable branches (after the bifurcation point) evolve not symmetrically around $I_{out1} = 100\,nA$. Also, the middle unstable branch increases after the bifurcation.

Thanks to this bifurcation diagram, we know that for the simulations in cadence we will need to use the following variables values $I_{thr} = 100\,nA$, $I_{in1} = I_{in2} = I_{lin} = 300\,nA$. We will still fix the input current to $300\,nA$ since this the adaptation was done just to see the wanted behaviour in Julia but was working without any issues in cadence.

It will be seen letter on that this second try of bifurcation diagram will correspond much more to what will be obtained in cadence.

0.2 2 neurons - Mutual excitation

Instead of inhibiting the other neurons, this time the neurons excite each other. The corresponding network and block diagram are given by figure 3 and 4.

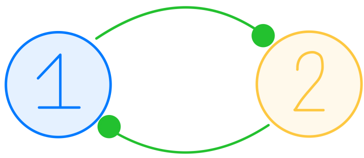


FIGURE 3 – Schematic view of the two neurons mutual excitation network

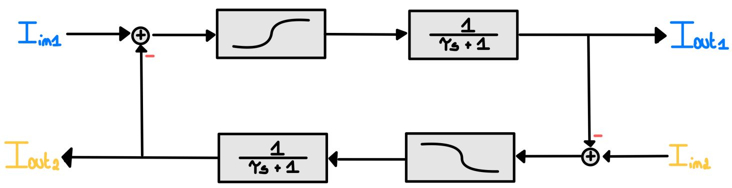


FIGURE 4 – Block diagram of the two neurons mutual excitation schematic

In opposition to the mutual inhibition case, each neurons receives as input its fixed input to which the output of the other neuron is added. The real input to each neuron is :

$$I'_{in1} = I_{gain} + I_{out2}$$
$$I'_{in2} = I_{gain} + I_{out1}$$

The bifurcation diagram of I_{out1} as a function of I_{gain} can be seen in figure 5.

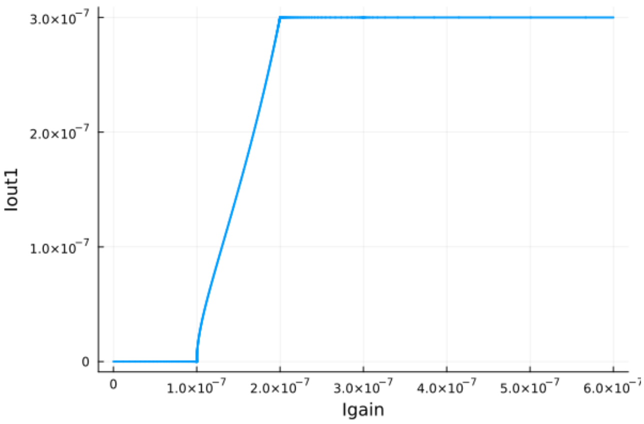


FIGURE 5 – Bifurcation diagram of I_{out1} as a function of I_{gain}

No bifurcation are obtained since the output current reaches much faster the saturation. Once the saturation is reached, obtaining a bifurcation is really unlikely. It is really unlikely since to obtain a bifurcation the system must be really sensible to small input changes and when being at saturation it will not be the case. For that reason, we let down the mutual excitable case and focus on the mutual inhibition case.

0.3 3 neurons case

Now that we have our basic case with two mutually inhibiting neurons, we will try to extend it with three neurons in a first place. The three mutually inhibiting neurons can be represented by :

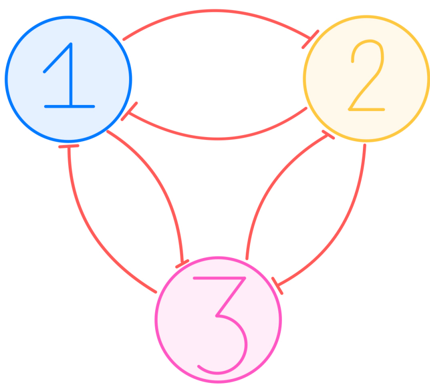


FIGURE 6 – Schematic view of the three neurons mutual inhibition network

The corresponding block diagram is :

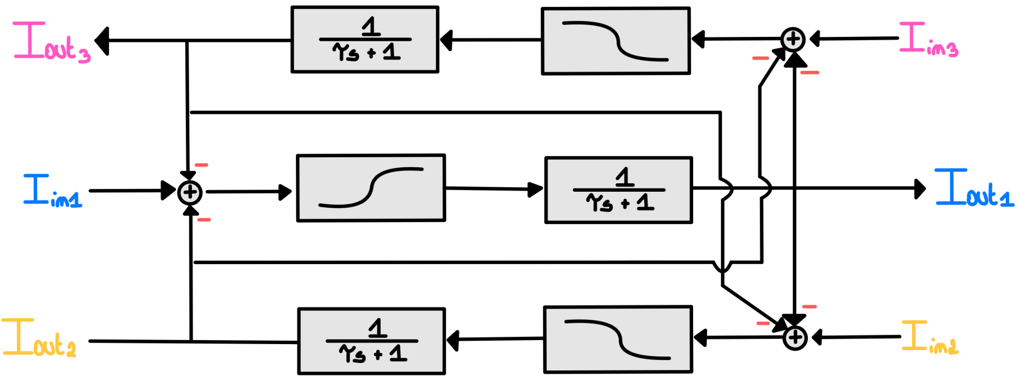


FIGURE 7 – Block diagram of the three neurons mutual inhibition schematic

The same idea as for the two neurons case is used. In other words, as input each neurons will have its fixed input current to which the output current of the other neurons will be subtracted. So, the input of the different neurons are :

$$I_{in1}' = I_{in1} - I_{out2} - I_{out3}$$
$$I_{in2}' = I_{in2} - I_{out1} - I_{out3}$$
$$I_{in3}' = I_{in2} - I_{out1} - I_{out2}$$

Once we tried to do the simulations we were confronted to some limitation of our model. This limitation comes from the fact that we could be confronted to some cases in which one of the $I_{in,i}'$ $(i = 1, 2, 3)$ could be negative. Since the system is solved using voltage, this input current is transformed into an input voltage using the diode-connected transistor for a pmos formula $(V_{P_{\text{diode}}}(I_{in}) = V_{dd} - \frac{U_T}{\kappa} \log\left(\frac{I_{in}}{I_0}\right))$. Since the input current is present in a logarithm it creates an error. Because the input currents could not be negative, we fixed their value at zero when they where negative. This lead to infinite input voltage with which the solver was loss.

On the other hand, it was working in cadence and providing satisfying results while using the parameters values from the two neurons case. For that reason, simulation were let down for this case and cases with more neurons since the same limitation was observed.

0.4 Stability at the bifurcation point

In the analysis three cases were considered, the 2, 3 and 5 neurons case. For each of those different cases, we consider each neurons inhibiting its neighbours. We want to see if there is a relation between the number of neurons and the bifurcation point.

For our analysis, we will use the really simple sigmoid equation with a low pass filter :

$$\dot{I}_{out,i} = -I_{out,i} + k \cdot \tanh\left(\bar{I}_{in,i} - \sum_{j \neq i} I_{out,j}\right)$$

Considering the 2 neurons case, the system of equations is :

$$\dot{I}_{out,1} = -I_{out,1} + k \cdot S\left(\bar{I}_{in,1} - I_{out,2}\right)$$
$$\dot{I}_{out,2} = -I_{out,2} + k \cdot S\left(\bar{I}_{in,2} - I_{out,1}\right)$$

To be able to compute the eigenvalues, we need to look at the steady state equations

$$0 = -I_{out,1} + k \cdot S\left(\bar{I}_{in,1} - I_{out,2}\right)$$
$$0 = -I_{out,2} + k \cdot S\left(\bar{I}_{in,2} - I_{out,1}\right)$$

Then, computing the Jacobian :

$$J_2 = \begin{bmatrix} -1 & -k \cdot \dot{S} \\ -k \cdot \dot{S} & -1 \end{bmatrix}$$

where $\dot{S}$ is the derivative of the sigmoid. The derivative of the sigmoid corresponds to the slope of that sigmoid. Considering the Jacobian at the bifurcation point means that we need to take $\bar{I}_{in,1} = \bar{I}_{in,2} = 300 \cdot 10^{-9}\,A$ and $I_{out,1} = I_{out,2} = 100 \cdot 10^{-9}\,A$ letting k as variable for the moment. For that reasons, we need to compute the slope of the sigmoid.

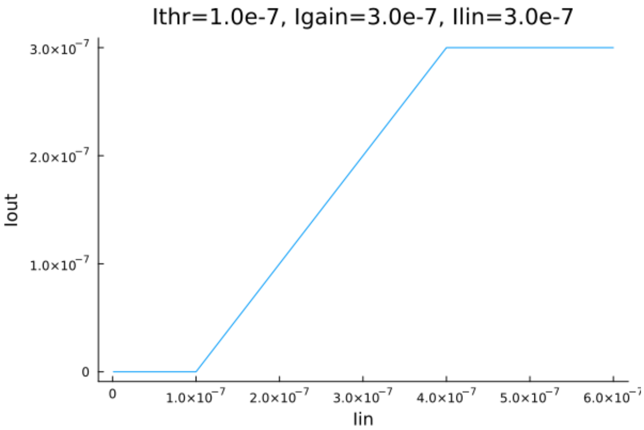


FIGURE 8 – Sigmoid in the 2 neurons case

Knowing that the bifurcation is at about $I_{in,1} = 300\,nA$, the slope around that point is computed giving $\dot{S} \approx \frac{1}{2}$. For that reason, the Jacobian becomes :

$$J_2 = \left.\begin{bmatrix} -1 & -k \cdot \frac{1}{2} \\ -k \cdot \frac{1}{2} & -1 \end{bmatrix}\right|_{\bar{I}_{in,1}=\bar{I}_{in,2}=300\cdot 10^{-9}\,A,\; I_{out,1}=I_{out,2}=100\cdot 10^{-9}\,A}$$

The Jacobian becomes singular for $k \approx 2$.

The same analysis can be done in the 3 neurons case for which the system of equations is of the form :

$$\dot{I}_{out,1} = -I_{out,1} + k \cdot S\left(\bar{I}_{in,1} - I_{out,2} - I_{out,3}\right)$$
$$\dot{I}_{out,2} = -I_{out,2} + k \cdot S\left(\bar{I}_{in,2} - I_{out,1} - I_{out,3}\right)$$
$$\dot{I}_{out,3} = -I_{out,3} + k \cdot S\left(\bar{I}_{in,3} - I_{out,1} - I_{out,2}\right)$$

The corresponding steady state equations are :

$$0 = -I_{out,1} + k \cdot S\left(\bar{I}_{in,1} - I_{out,2} - I_{out,3}\right)$$
$$0 = -I_{out,2} + k \cdot S\left(\bar{I}_{in,2} - I_{out,1} - I_{out,3}\right)$$
$$0 = -I_{out,3} + k \cdot S\left(\bar{I}_{in,3} - I_{out,1} - I_{out,2}\right)$$

Then, computing the Jacobian :

$$J_3 = \begin{bmatrix} -1 & -k\cdot\dot{S} & -k\cdot\dot{S} \\ -k\cdot\dot{S} & -1 & -k\cdot\dot{S} \\ -k\cdot\dot{S} & -k\cdot\dot{S} & -1 \end{bmatrix}$$

As in the two neurons case, considering the Jacobian at the bifurcation point means that we need to take $\bar{I}_{in,1} = \bar{I}_{in,2} = \bar{I}_{in,3} \approx 300 \cdot 10^{-9}\,A$. Looking again at the sigmoid :

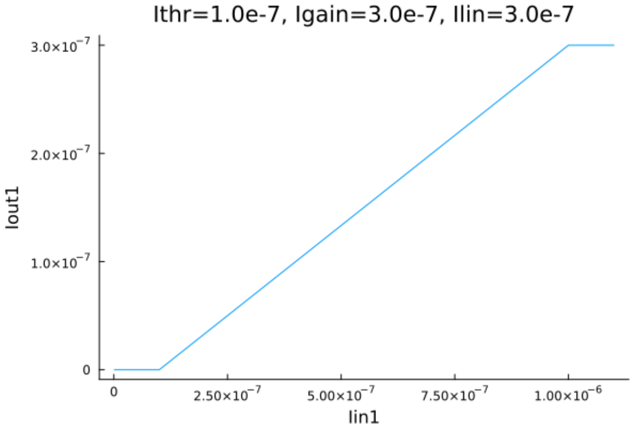


FIGURE 9 – Sigmoid in the 3 neurons case

The slope of the sigmoid around I_{in} is this time about $\frac{1}{3}$ leading to $\dot{S} = \frac{1}{3}$. It can we remarked that the slope is more or less the same in the intervalle $I_{in} \in [1 \cdot 10^{-7}, 10 \cdot 10^{-7}]$ since we are in the linear part of the $I_{in} - I_{out}$ characteristic. So, the Jacobian evaluated at that point is :

$$J_3 = \begin{bmatrix} -1 & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} \\ -k\cdot\frac{1}{3} & -1 & -k\cdot\frac{1}{3} \\ -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -1 \end{bmatrix}\Bigg|_{\bar{I}_{in,1}=\bar{I}_{in,2}=\bar{I}_{in,3}=300\cdot 10^{-9}\,A},$$

The Jacobian this time becomes singular for $k = 3$.

It can be observed that depending on the case, the idealized parameter k representing I_{gain} to make the Jacobian singular depends on the number of neurons. In fact, it depends on the number of inhibiting inputs each neurons receives. Because the value of k vary to obtain a zero eigenvalue, it would lead to a different point for the bifurcation to happen.

Considering this time the 5 neurons cases in which each neurons is connected to two others neurons as represented in figure ??. The corresponding system of equation is given by :

$$\begin{aligned}
\dot{I}_{out,1} &= -I_{out,1} + k \cdot S\left(\bar{I}_{in,1} - I_{out,2} - I_{out,5}\right) \\
\dot{I}_{out,2} &= -I_{out,2} + k \cdot S\left(\bar{I}_{in,2} - I_{out,1} - I_{out,3}\right) \\
\dot{I}_{out,3} &= -I_{out,3} + k \cdot S\left(\bar{I}_{in,3} - I_{out,1} - I_{out,4}\right) \\
\dot{I}_{out,2} &= -I_{out,4} + k \cdot S\left(\bar{I}_{in,4} - I_{out,3} - I_{out,5}\right) \\
\dot{I}_{out,5} &= -I_{out,5} + k \cdot S\left(\bar{I}_{in,5} - I_{out,1} - I_{out,4}\right)
\end{aligned}$$

The corresponding steady states equations are :

$$
\begin{aligned}
0 &= -I_{out,1} + k \cdot S\left(\bar{I}_{in,1} - I_{out,2} - I_{out,5}\right) \\
0 &= -I_{out,2} + k \cdot S\left(\bar{I}_{in,2} - I_{out,1} - I_{out,3}\right) \\
0 &= -I_{out,3} + k \cdot S\left(\bar{I}_{in,3} - I_{out,1} - I_{out,4}\right) \\
0 &= -I_{out,4} + k \cdot S\left(\bar{I}_{in,4} - I_{out,3} - I_{out,5}\right) \\
0 &= -I_{out,5} + k \cdot S\left(\bar{I}_{in,5} - I_{out,1} - I_{out,4}\right)
\end{aligned}
$$

Then, computing the Jacobian leads to :

$$
J_5 = \begin{bmatrix}
-1 & -k\cdot\dot{S} & -k\cdot\dot{S} & -k\cdot\dot{S} & -k\cdot\dot{S} \\
-k\cdot\dot{S} & -1 & -k\cdot\dot{S} & -k\cdot\dot{S} & -k\cdot\dot{S} \\
-k\cdot\dot{S} & -k\cdot\dot{S} & -1 & -k\cdot\dot{S} & -k\cdot\dot{S} \\
-k\cdot\dot{S} & -k\cdot\dot{S} & -k\cdot\dot{S} & -1 & -k\cdot\dot{S} \\
-k\cdot\dot{S} & -k\cdot\dot{S} & -k\cdot\dot{S} & -k\cdot\dot{S} & -1
\end{bmatrix}
$$

To be able to determiner the derivative of the sigmoid $\dot{S}$, we need to look at the input-output current characteristic of the 5 neurons case :

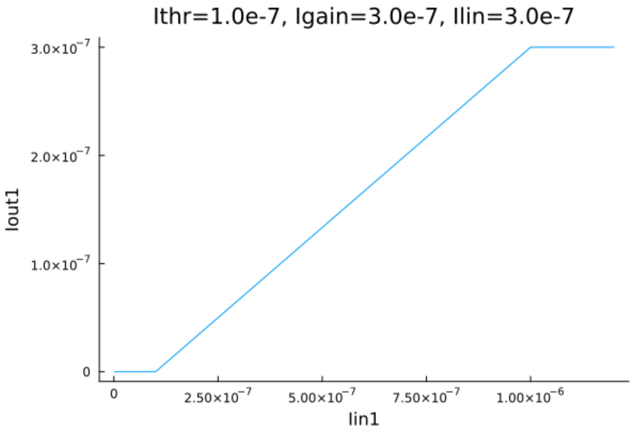


FIGURE 10 – Sigmoid in the 5 neurons case

The slope of the linear region is $\frac{1}{3}$ as in the 3 neurons case. For that reason, the Jacobian becomes :

$$
J_5 = \begin{bmatrix}
-1 & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} \\
-k\cdot\frac{1}{3} & -1 & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} \\
-k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -1 & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} \\
-k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -1 & -k\cdot\frac{1}{3} \\
-k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -k\cdot\frac{1}{3} & -1
\end{bmatrix}
$$

The value of k making the Jacobian singular is $k = 3$

So, the bifurcation point do not depend on the number of neurons but more on the number of inhibiting inputs a neurons receives. It is for that reason that for the 2 neurons case the idealize parameter representing I_{gain} can be smaller ($k = 2$) than in the 3 or 5 neurons case ($k = 3$) to make the corresponding Jacobian matrix singular. The reason is that when there is only 2 neurons each neurons is only inhibited by 1 neuron. On the other hand, in the 3 and 5 neurons case, each neurons is inhibited by both of its neighbours and thus need a larger gain. For that reason, the bifurcation point should be the same for a 3 or 5 neurons circuit but different from the 2 neurons circuit.

1 Cadence Implementation

1.1 Sigmoid Circuit Implementation

After getting the results of the Julia implementation, we could move to Cadence implementation. We first need to slightly modify the sigmoid circuit we learned in practical lessons to make it suitable for our design. We therefore added an input pin and an output pin at the gate of current mirrors (figure 11).

The voltage supply will be constant in the entire circuit and will be fixed to $V_{dd} = 1.8V$.

As you can observe, the gate voltage V_{thr}, V_{lin}, V_0 are controlled with current mirrors so we can control it rather in current or the reasons mentioned earlier.

We also modified the transistors dimensions. These will be fixed to the same value for the whole circuit. We have the width of the transistor fixed to $2\mu m$ and the length to $1\mu m$, providing a safe margin against the Early effect. It's a behavior that results from the modulation of the base width by the collector-base voltage, leading to changes in collector current and affecting the transistor's performance in circuits.

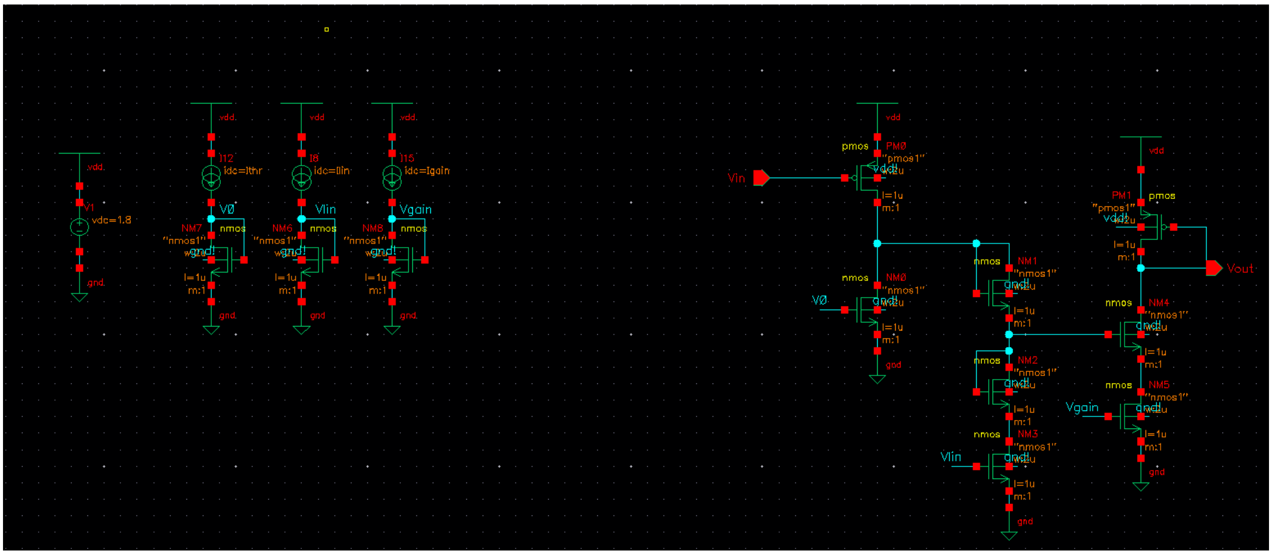


FIGURE 11 – Sigmid Circuit Implementation on Cadence

In order to make the circuit more readable, one creates a symbol representing this circuit (plot 12). The input and output pins correspond to the ones shown on figure 11.

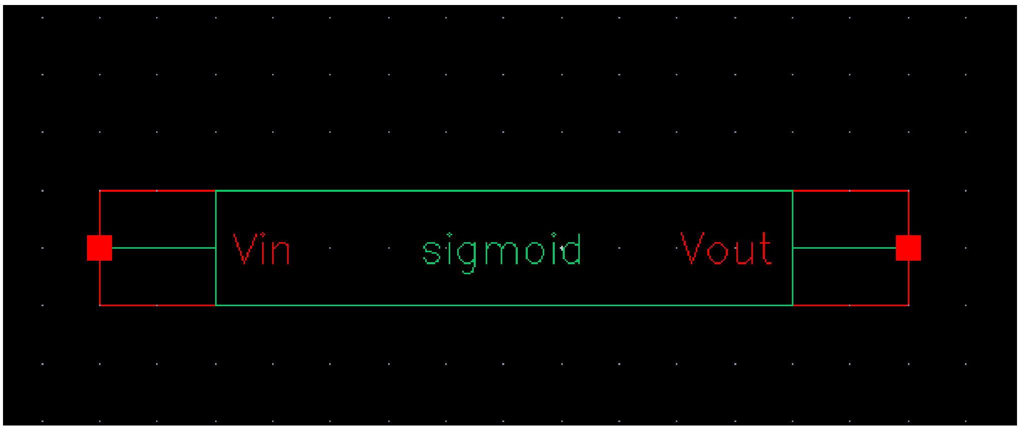


FIGURE 12 – Sigmoid Symbol

1.2 2 neuron network

In this part of the project, we aim to explain how from simple sigmoid we managed to interconnect several block and to study their global behaviour.

1.2.1 First Try

The first (functionnal) circuit we got was the one shown on 13. Let us try to explain a bit how this circuit basically works.

The output of each sigmoid is connected to another pmos to form a current mirror. Therefore, the output current of the sigmoid is copied and it goes to the other sigmoid block. At that point, one can apply the first kirchoff law at the drain of pmos 2 at the entrace of the second sigmoid block :

$$I_{s\to d} = I_{in2} - I_{out1}$$

and for PM0 at the entrance of the first sigmoid block :

$$I_{s\to d} = I_{in1} - I_{out2}$$

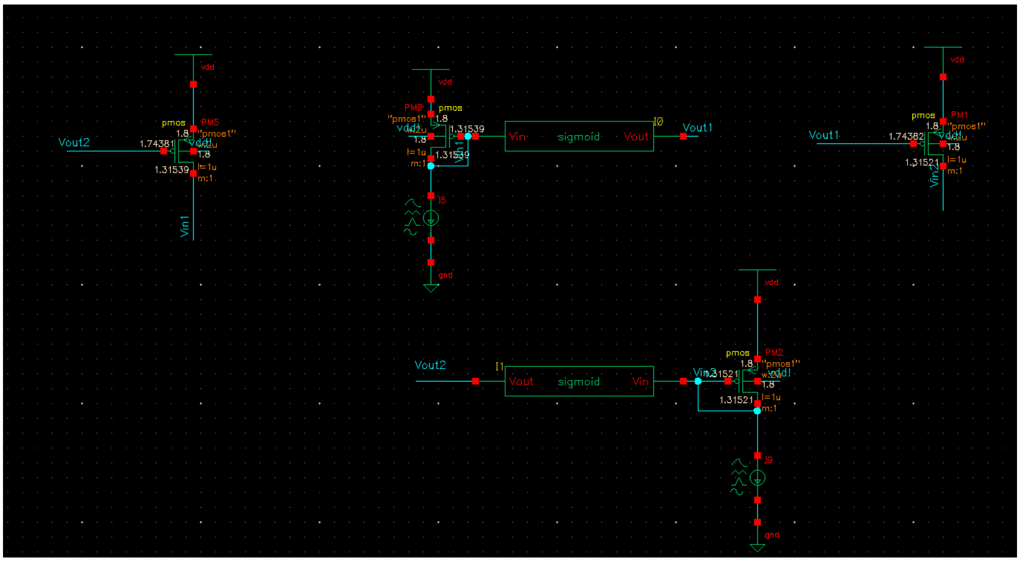


FIGURE 13 – First 2 sigmoid circuit

Now we can finally try what we have been searching for in Julia : the point where one neuron will take over the other. We fix $I_{thr} = 100nA$, $I_{lin} = 300nA$ and $I_{in1} = I_{in2} = 300nA$. The julia simulations told us that the bifurcation point should be at $I_{gain} = 300nA$. We therefore perform a DC simulation using I_{gain} as a bifurcation parameter. As you can see on plot 14, this is definitely not what we expected. The two output currents of the sigmoids blocks are the same for any I_{gain} even if we expected bifurcation.

What we therefore decided to do is to include a slight asymmetry in the input currents. Let us repeat the same experiment for $I_{in1} = 301nA$ and $I_{in2} = 300nA$, results can be seen on 15, the red line is the output of the first block, the one with the slightly larger input, and the blue one the output current of the block with the smallest input. Now, we have something closer than what we expected. But it's not yet perfect since the bifurcation is not a squared pitchfork like in Julia and we want to get closer from it. Notice that the DC solver of cadence stays on the unstable branch equivalent in Julia if no asymmetry is given.

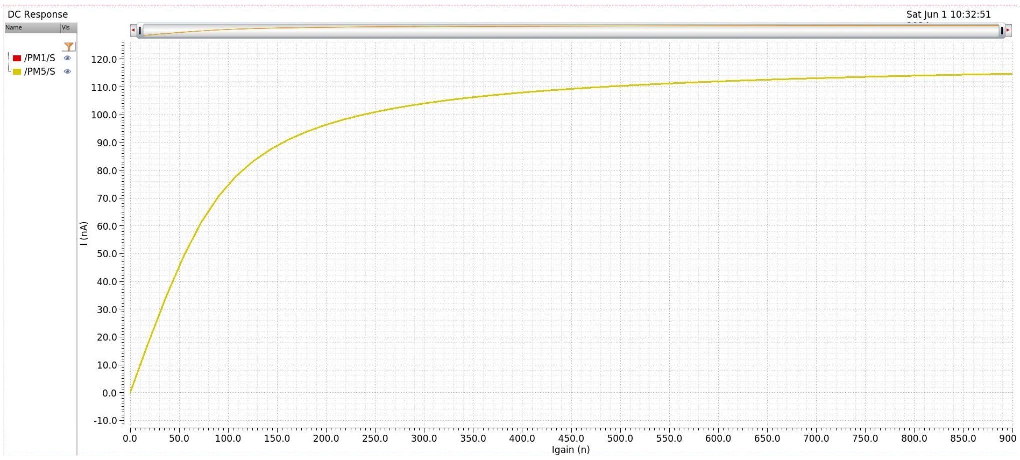


FIGURE 14 – Simulation for $I_{in1} = I_{in2} = 300nA$

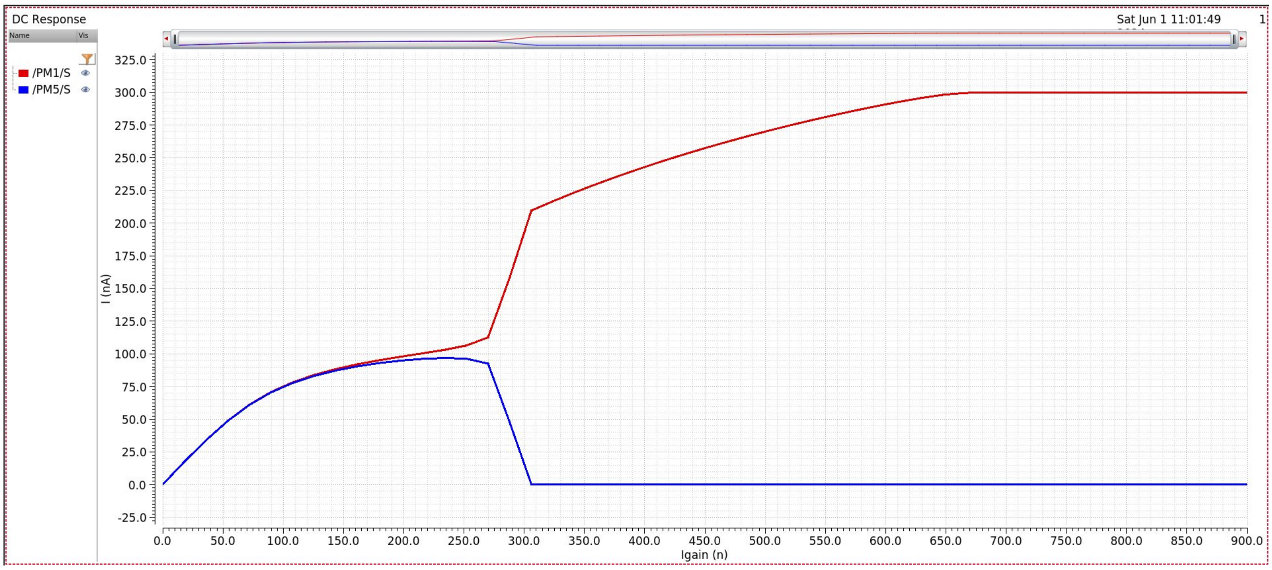


FIGURE 15 – Simulation for $I_{in1} = 301nA$,$I_{in2} = 300nA$

We will not be able to solve that problem with this circuit, this is why we move to the next part where we search for a solution that needs less asymmetry to bifurcate.

1.2.2 Second try : Adding transmission gates

We now decide to insert at the output of the sigmoid circuits transmission gates (figure 16). A transmission gate is an electronic component used in digital circuits, made up of a parallel combination of an NMOS and a PMOS transistor. It's essentially a switch that can pass both high and low voltage levels with minimal resistance and distortion.

— When the control input is high (1.8 Volt in our application) :
  1. The gate of the NMOS transistor is at a low voltage.
  2. The gate of the PMOS transistor, controlled by an inverter, is at a high voltage.
  3. This setup ensures that the gate-source voltage of the NMOS is negative, and the gate-source voltage of the PMOS is positive, preventing both transistors from conducting. Thus, the transmission gate is turned off.

— When the control input is low (0 Volts here) :
  1. The gate of the NMOS transistor is at a high voltage.

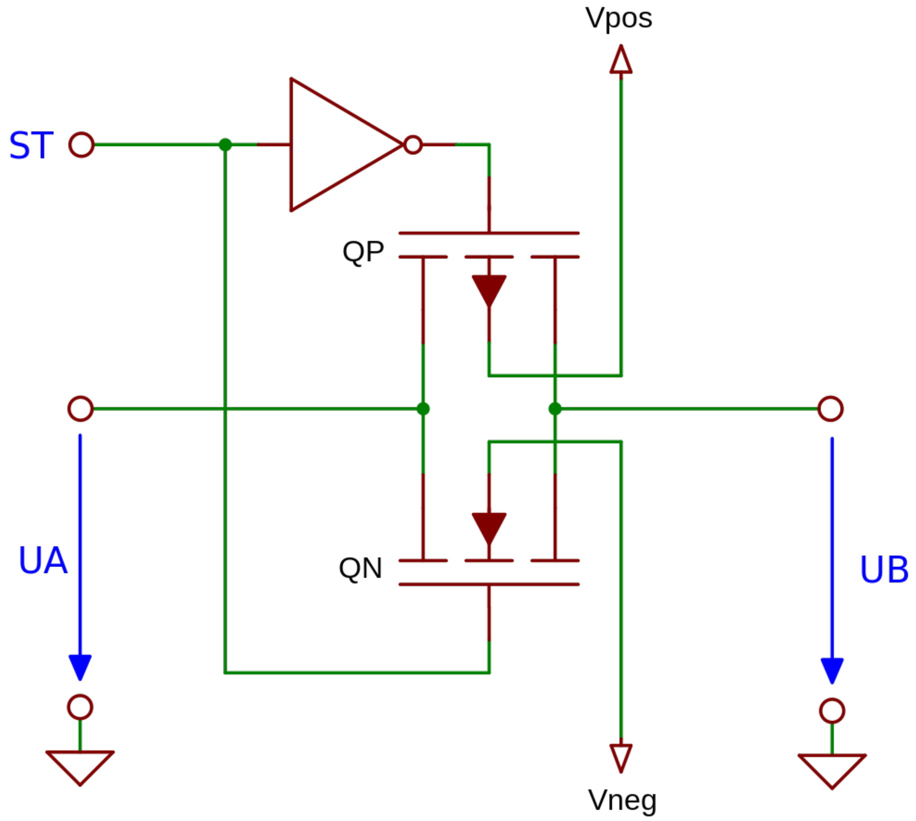


FIGURE 16 – Transmission Gate Schematic

2. The gate of the PMOS transistor, again controlled by the inverter, is at a low voltage.
3. This configuration ensures that the gate-source voltage of the NMOS is positive, and the gate-source voltage of the PMOS is negative, allowing both transistors to conduct. The transmission gate is then turned on, allowing signals to pass through.

Here we will use it as a low resistance (when conducting) analog switch, that can connect or disconnect different parts, allowing signal routing. In order to use such gates, one needs to find a way to complement the control signal for the gate voltage of the Pmos. This can be done thanks to a simple not gate (figure 17). It takes a simple binary input and outputs the inverse of its input. We created a dedicated block for it in cadence.

We also added a low pass filter to make it more robust against potential high frequency parasitic signals at the output of the sigmoids, we chose a 10kHz cutoff frequency.

Finally, one can analyze the final circuit shown on figure 18. We can now try to simulate it like we did for the previous one. If we do not give any asymmetry, the same behaviour can be observed than for the previous circuit. However, if now one has $I_{in1} = 300.01nA$ and $I_{in2} = 300nA$ so the asymmetry is a hundred times lower than the previous circuit (the variability of the circuit is slightly increased when one adds components, making it less robust to asymmetry), we can see this circuit bifurcating (plot 19). One can observe that we are now much closer to what we expected since the bifurcation is much more square-like and closer to 300nA. To confirm what we just discovered, let us perform some transient analysis. A transient analysis is a plot as a function of time. The first one will be for $I_{gain} = 100nA$, the second for $I_{gain} = 500nA$. Thus the first is in the zone where there is no bifurcation, and the second one is one asymmetry among solutions. Be carefull that the bifurcation point on the dc analyzis plot depends of the asymmetry. Indeed, the asymmetry is smaller on graph 19 than on 20 where

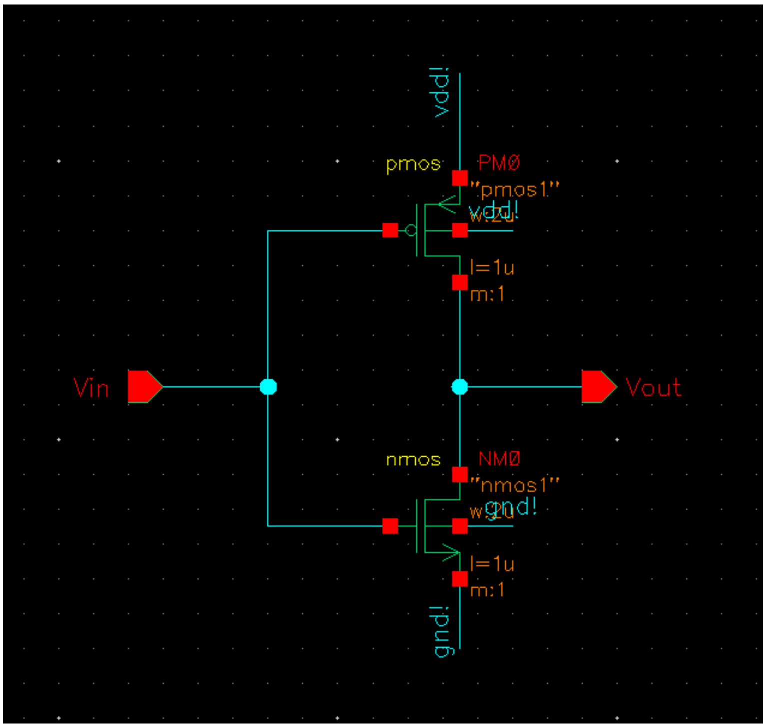


FIGURE 17 – Not Gate Schematic

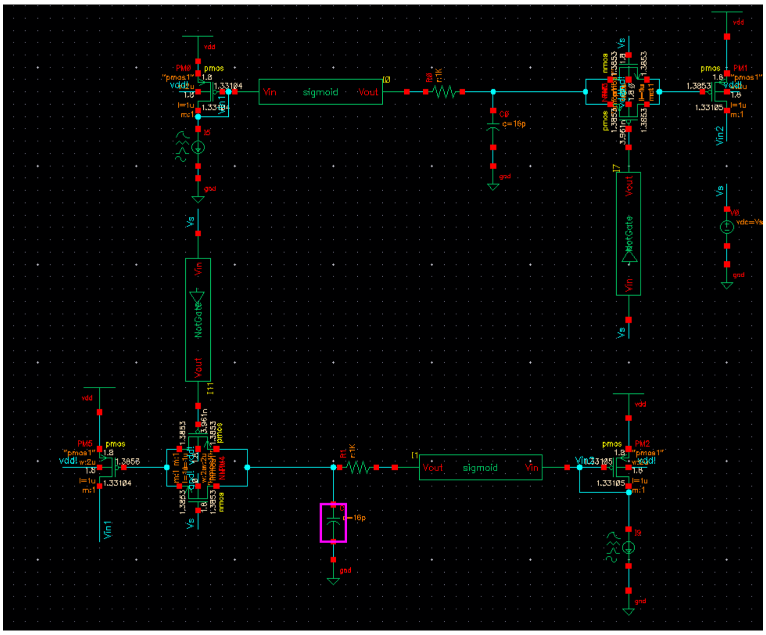


FIGURE 18 – Improved 2 sigmoids Schematic

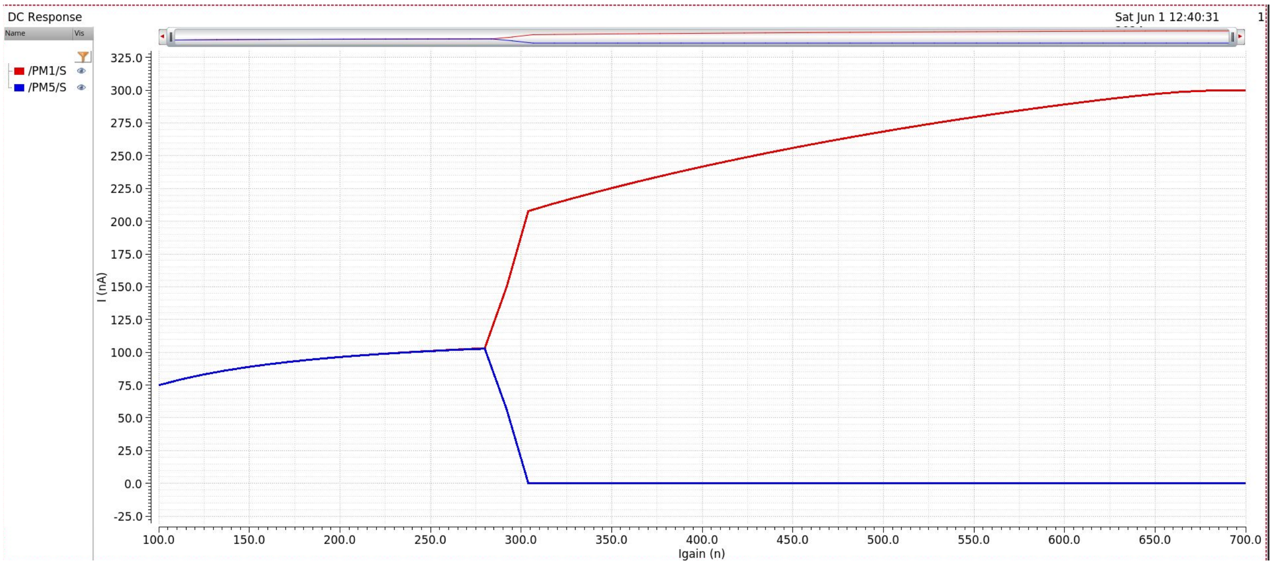


FIGURE 19 – Simulation with transmission gates for $I_{in1} = 300.01nA$ and $I_{in2} = 300nA$

it seems that the bifurcation happens "earlier". One will need to be carefull for the rest of the project where sometimes we may need to impose larger asymmetry to see bifurcation and complements its dc analyzis with transient analyzis that are less sensitive to this phenomenon.

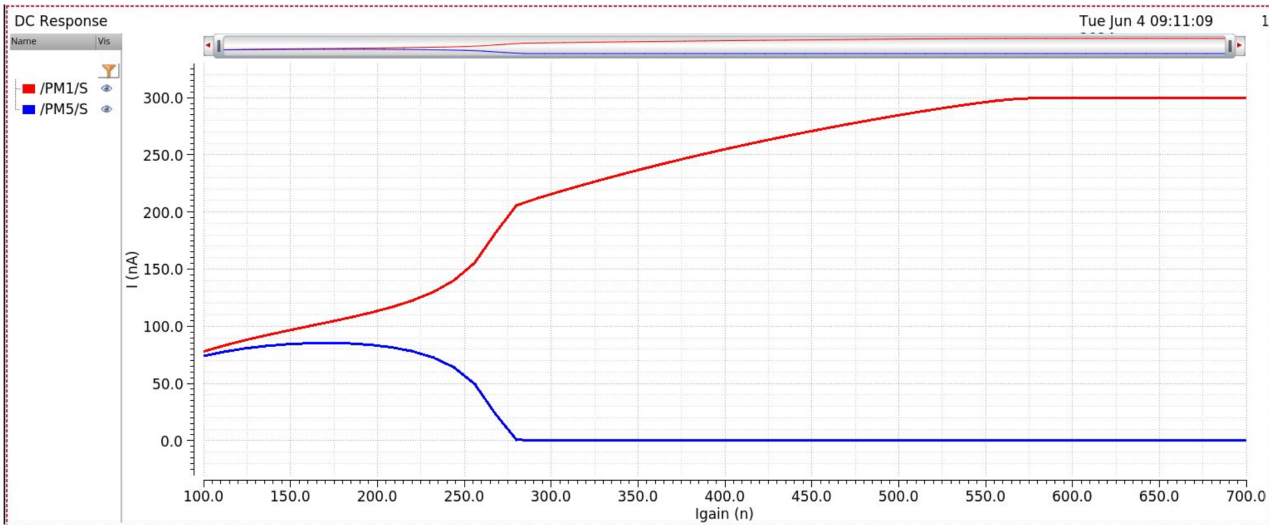


FIGURE 20 – Simulation with transmission gates for $I_{in1} = 310nA$ and $I_{in2} = 300nA$

Indeed, on figure 21, you can see the value is the same for both currents (small difference due to the asymmetry we impose). However, on plot 22, the values diverge, showing the expected behaviour. Using this type of reasoning allowed to confirm the bifurcation point is indeed for $I_{gain} = 300nA$

One can now perform a transient analysis to show that the highest output current corresponds to the highest input current. We take to input currents to 300nA and we add a sine component of some amplitude to the input current, the system shows some hysteresis since sufficient change in the input is required to change the neuron that thrives. As can be seen on figure 23, the sine component input is not big enough to cause a change in the winner but on figure 24, it is sufficiently big to cause a change in the winner.

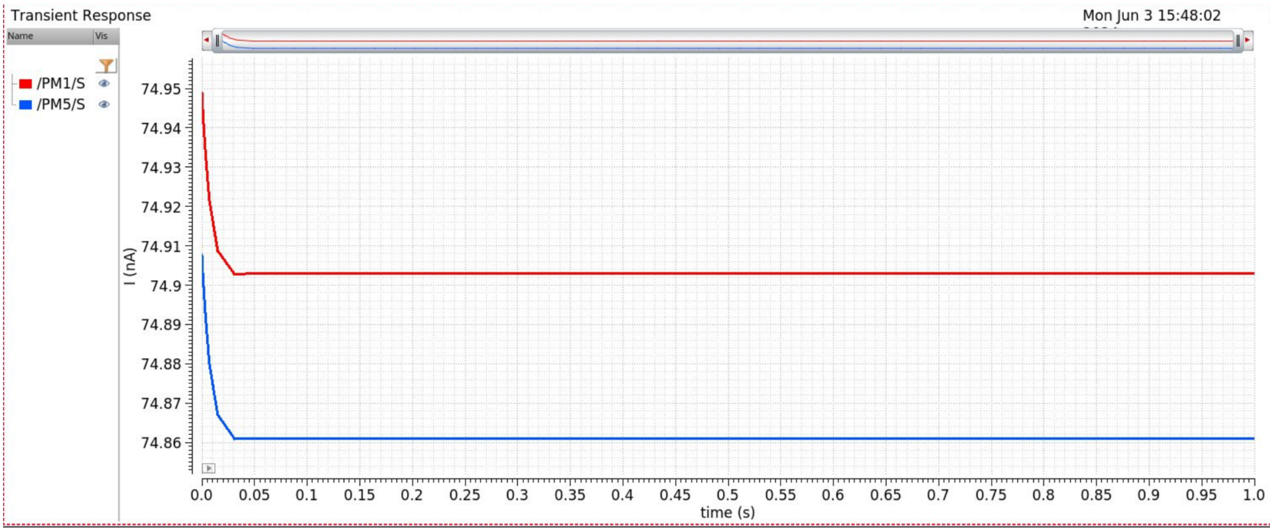


FIGURE 21 – Transient simulation with $I_{gain} = 100nA$

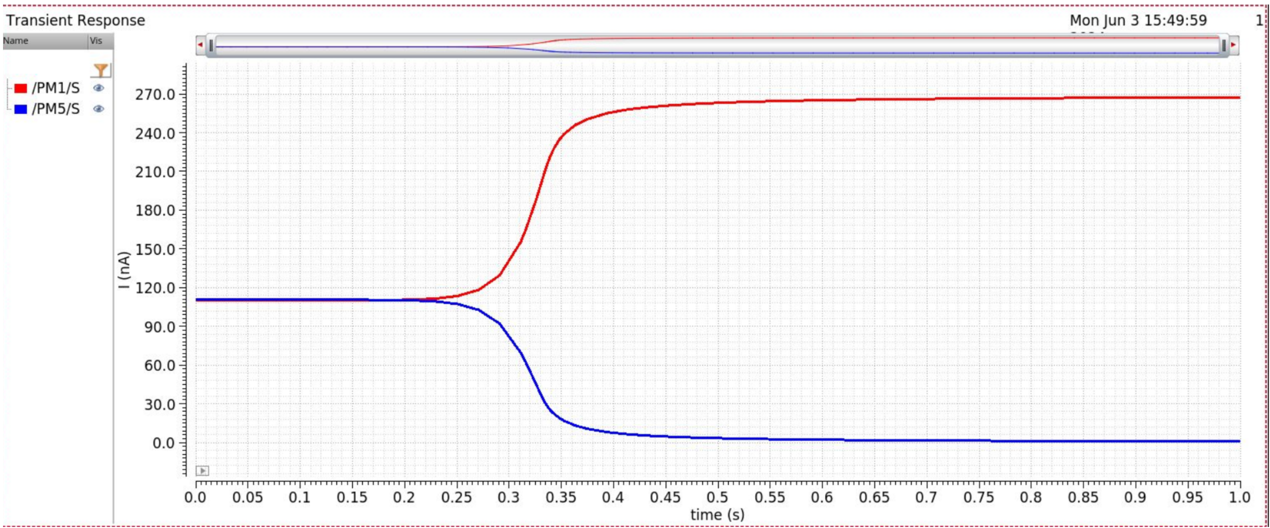


FIGURE 22 – Transient simulation with $I_{gain} = 500nA$

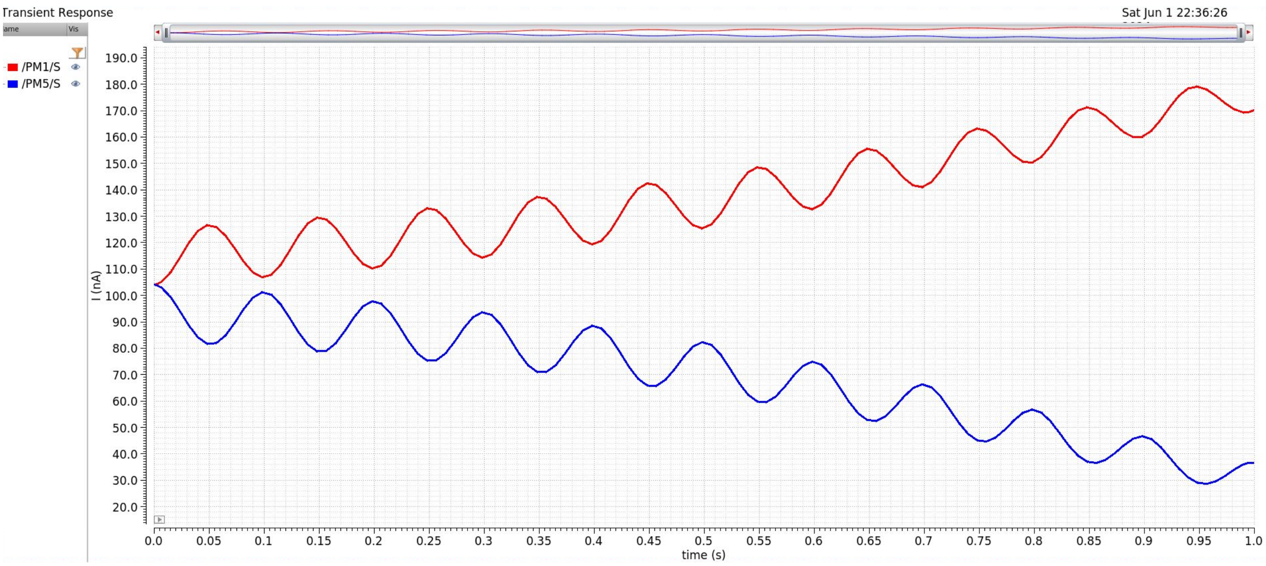


FIGURE 23 – Transient simulation with both dc component of input current of 300nA and 5pA of sine component

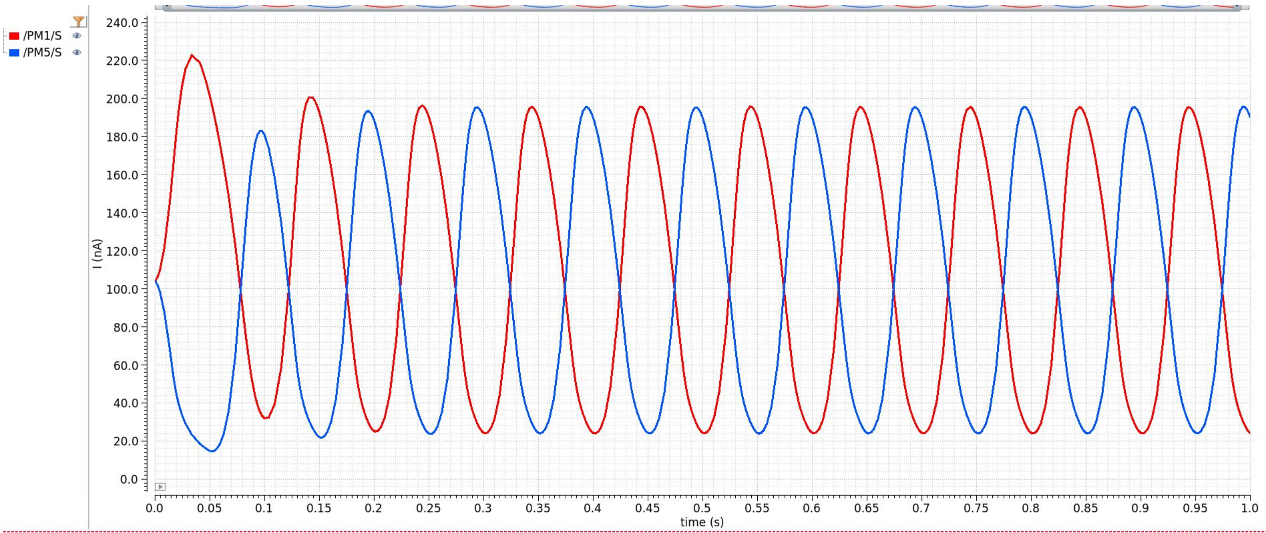


FIGURE 24 – Transient simulation with both dc component of input current of 300nA and 50pA of sine component

1.3 3 neurons circuit

Now that we have our building block with the 2 neurons circuits, one can try to move to 3 neurons that inhibit each other like represented on 25. One can observe that one should just use a third building block and use a second current mirror to each output (therefore the use of current mirror is now evident to replicate the current from some reference).

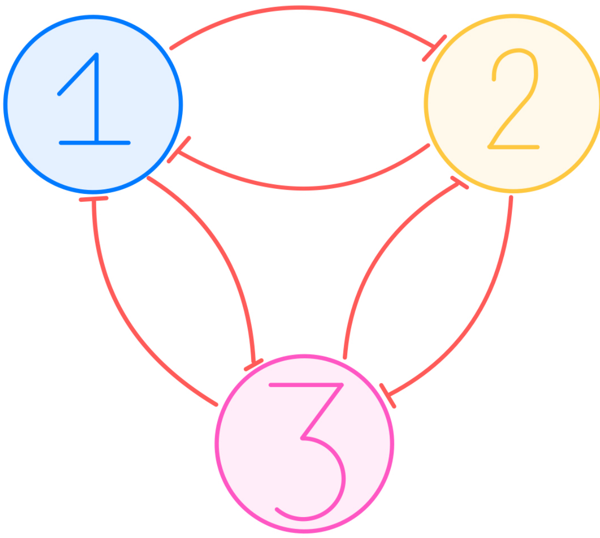


FIGURE 25 – 3 neurons schematic

The implementation of the circuit in cadence is given by figure 27.

We direclty made use of transmission gates. This circuit allowed to verify what we saw earlier, so if there is a tiny asymmetry between the 3 neurons, bifurcation will occur and one will observe that the output current of the corresponding neuron is higher while the two other neurons are shut down (figure 27).

The bifurcation point is now for lower gain current than before. Actually, it depends on the number of neurons one single neuron inhibits. So if we take back the same model (3 neurons) and that we set a control voltage of 0V for each transmission gate which is part of V_{21}, V_{32}, V_{13},

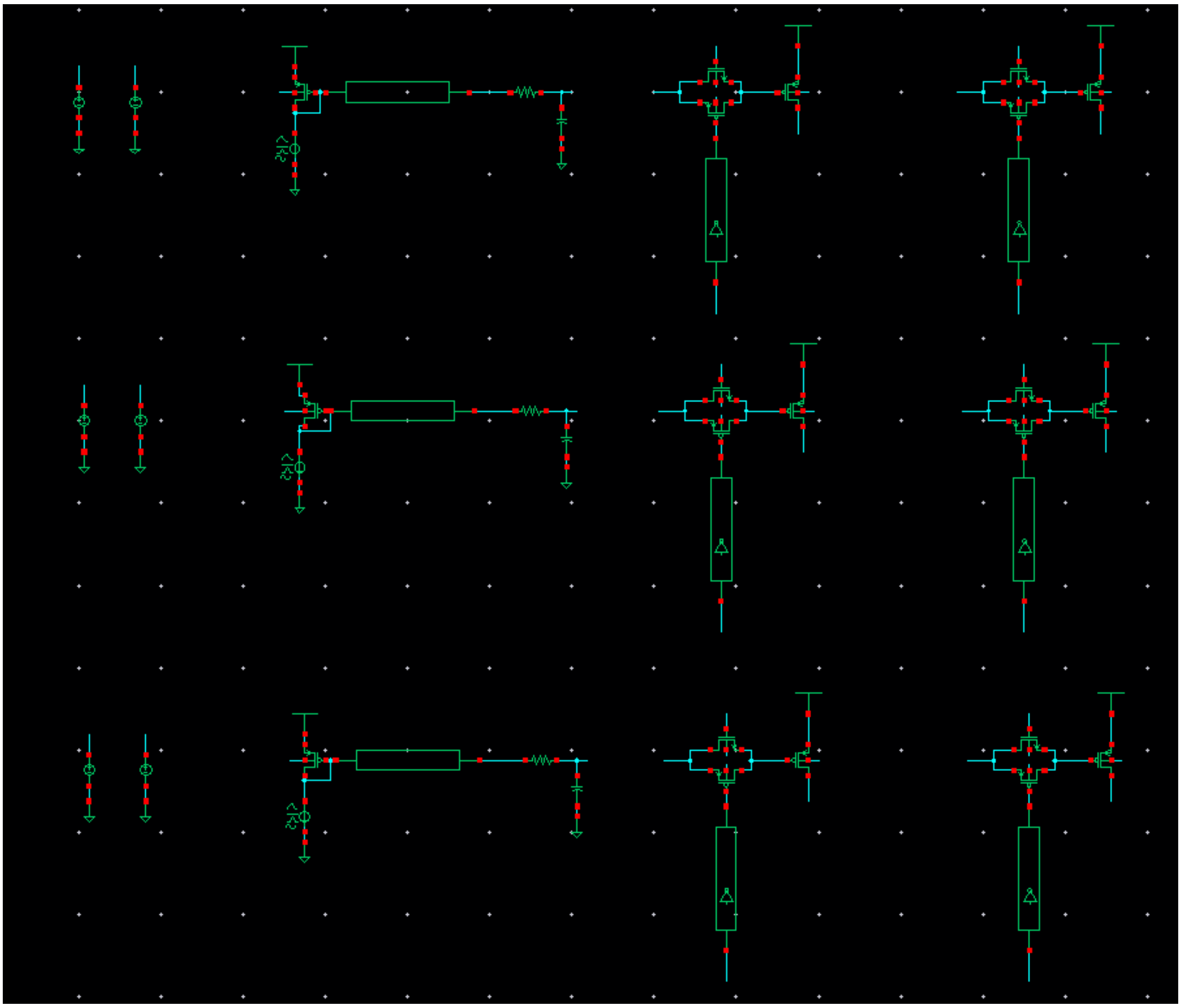


Figure 26 – 3 neurons implementation on Cadence

we see that the bifurcation point is now at the same place than the two neuron case. It's shown using transient analyzis rather than DC. Transient analyzis showed that now the bifurcation point is around $150nA$ when the neuron inhibits two of its neighbors rather than one(in one case it was 300nA) . The neuron that thrives is the one with slightly higher initial condition. Before bifurcation, the isotropy subgroup of state vector x is $\mathbf{S_3}$ and after bifurcation it's $\mathbf{S_1} \times \mathbf{S_2}$. Indeed, the two other neurons are merged in the green curve since they are equally inhibited with equal input currents.

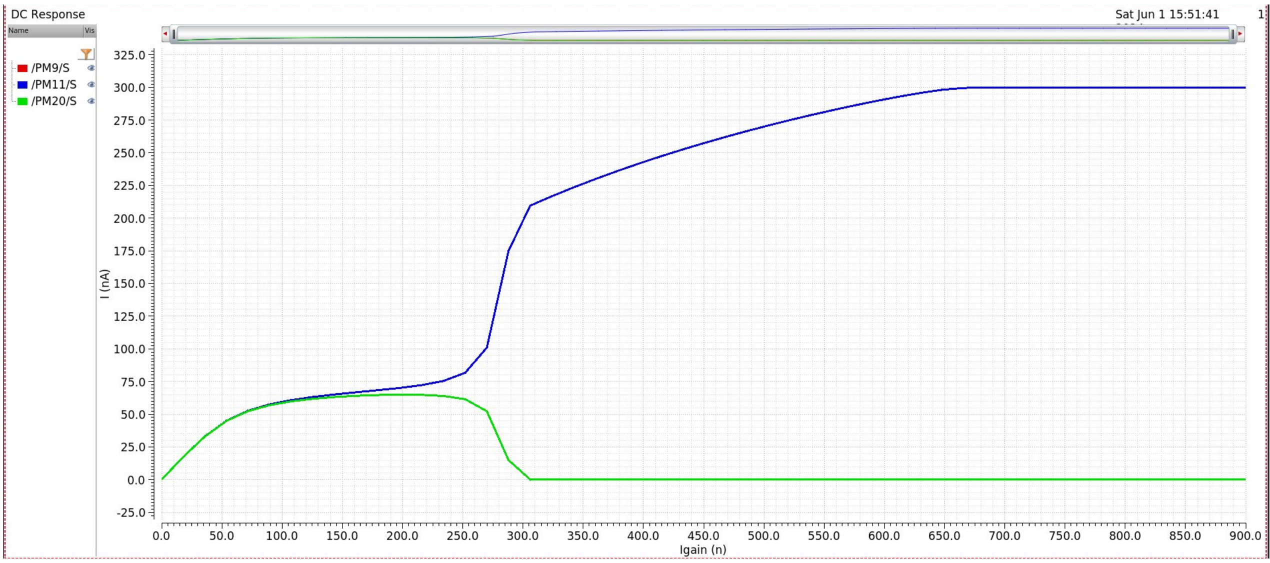


Figure 27 – 3 neurons bifurcation diagram in cadence

1.4 5 Neurons Circuit

This final circuit was made in 2 steps. In the first part, the circuit was $\mathbf{C_2} - equivariant$ while in the second step, it was $\mathbf{S_5} - equivariant$, which changed radically the behaviour of these circuits. Let's explain a bit these notions of equivariance applied to circuits and permuation matrices.

Consider the two following situations :

- A neuron network consisting of 5 neurons inhibiting their respective neighbors.
- A neuron network consisting of the same 5 neurons, but where the boundaries neuron don't inhibit each other anymore.

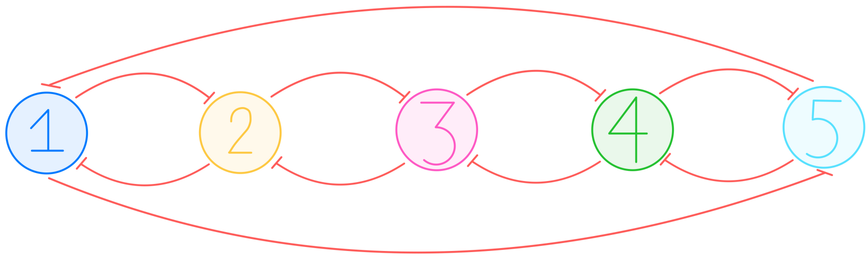


Figure 28 – Complete Model

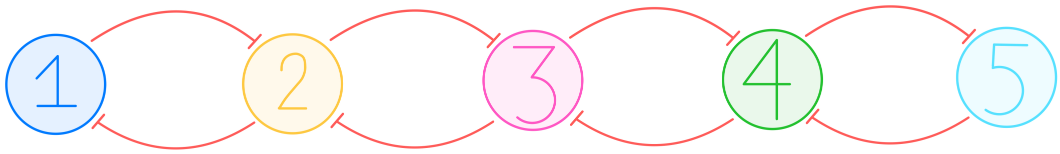


Figure 29 – Incomplete Model

Even though these networks might look very similar, let us show that their equivariant group are radically disimilar.

We begin our analysis with the incomplete model.

Let us note the group element $(x\,y)$ the element that permutes element x and y. Therefore, we note the system $f(\underline{x}) = \{1, 2, 3, 4, 5\}$ the system modeling our network. Let us consider $\sigma \in \mathbf{S_5}$ so that $\sigma = (2\,4)$. Taking its action on the system gives :

$$\sigma.f(\underline{x}) = \{1, 4, 3, 2, 5\}\,.$$

The figure 30 shows that the network is therefore absolutely different than before applying the simulation (because the two neurons on the left don't inhibit each other and the same for the two last ones).

Now one can consider applying the group element $\theta =< (1\,5), (2\,4) >$ using composition of function as a law of composition to the network and observe that this time the network is equivalent to the initial one (figure 31). Using this information, one has that $\theta \cdot \theta = i$ where i is the identity element of $\mathbf{S_5}$, $\theta^{-1} = \theta$ so it is straightforward that the system is $\mathbf{C_2} - equivariant$. This finite group has 2 elements $< e, g >$ where e is the identity and g the generator such that $g.g = e$, chararcterized by the Cayley Table :

$$\left[\begin{array}{c|cc} \cdot & e & g \\ \hline e & e & g \\ g & g & e \end{array}\right]$$

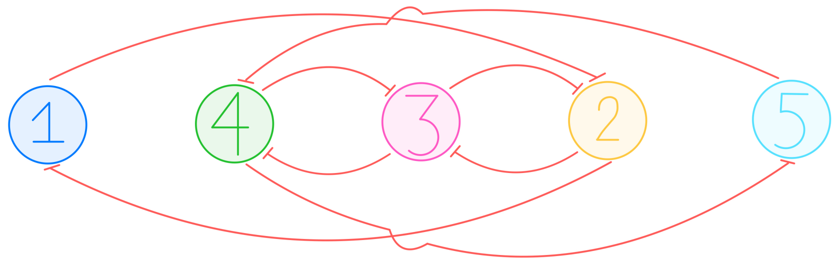


Figure 30 – Model after applying the permutation

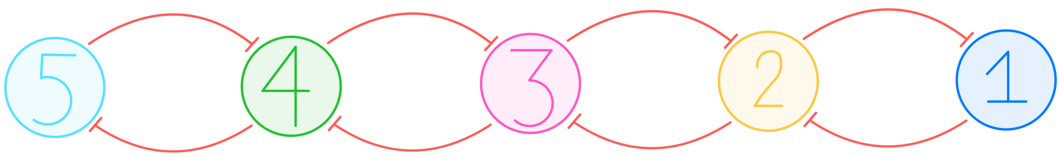


Figure 31 – Model after applying the second permutation

Let us now try to understand why θ is the generator of g while σ is not.

We will model the system using a state matrix where $a_{i,j}$ is :
— 0 if there is no connection from neuron i to neuron -1
— -1 if there is an inhibitory connection from neuron i to neuron j
— +1 if there is an excitatory connection from neuron i to neuron j.
Each neuron x_i can therefore gather all the connection that starts from it, with x_{i_j} denoting connection from neuron i to j.

Therefore we have the following :
— $(x_1)^T = [0, -1, 0, 0, 0]$
— $(x_2)^T = [-1, 0, -1, 0, 0]$
— $(x_3)^T = [0, -1, 0, -1, 0]$
— $(x_4)^T = [0, 0, -1, 0, -1]$
— $(x_5)^T = [0, 0, 0, -1, 0]$

So the state matrix is given by $A(\underline{x}) = [x_1, x_2, x_3, x_4, x_5]$. So $A(\underline{x}) = \begin{pmatrix} 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 \end{pmatrix}$ which is indeed symmetrical of course.

Now consider the permutation matrix Σ which is the isomorphism of σ, ie. the permutation matrix swapping element 2 and 4, it is $\Sigma = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$ Remember the definition of equivariance with respect to a group element γ : $\gamma.f(x) = f(\gamma.x)$

So now we have :
— $(\Sigma \cdot x_1)^T = [0, 0, 0, -1, 0]$
— $(\Sigma \cdot x_2)^T = [-1, 0, -1, 0, 0]$
— $(\Sigma \cdot x_3)^T = [0, -1, 0, -1, 0]$
— $(\Sigma \cdot x_4)^T = [0, 0, -1, 0, -1]$
— $(\Sigma \cdot x_5)^T = [0, -1, 0, 0, 0]$

so $A(\Sigma \cdot x) = \begin{pmatrix} 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 \\ 0 & -1 & 0 & -1 & 0 \\ -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \end{pmatrix}$

while $\Sigma \cdot A(x) = \begin{pmatrix} 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 \\ 0 & -1 & 0 & 0 & 0 \end{pmatrix}$

So one can observe $\Sigma \cdot A(x) \neq A(\Sigma \cdot x)$.

Now let us have Θ the isomorphism of θ so $\Theta = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$ which swaps columns 1 and 5 as well as columns 2 and 4. We now have :

— $(\Theta \cdot x_1)^T = [0, 0, 0, -1, 0]$
— $(\Theta \cdot x_2)^T = [0, 0, -1, 0, -1]$
— $(\Theta \cdot x_3)^T = [0, -1, 0, -1, 0]$
— $(\Theta \cdot x_4)^T = [-1, 0, -1, 0, 0]$
— $(\Theta \cdot x_5)^T = [0, -1, 0, 0, 0]$

so

$$A(\Theta \cdot x) = \begin{pmatrix} 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 \\ 0 & -1 & 0 & -1 & 0 \\ -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \end{pmatrix}$$

and

$$\Theta \cdot A(x) = \begin{pmatrix} 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 \\ 0 & -1 & 0 & -1 & 0 \\ -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \end{pmatrix}$$

So the system is indeed $\Theta - \textit{equivariant}$ which is logic since it's the isomorphism of θ and the system is $\theta - \textit{equivariant}$. Remember that Θ is the isomorphism of θ which is the generator of the cyclic group. Note that adding for example one neuron left and one right without linking the two boundaries would lead the group to still be $\mathbf{C_2} - \textit{equivariant}$ the difference would be that i would be the $(7 \cdot 7)$ identity matrix and $g =< (1\,7), (2\,6), (3\,5) >$, g's isomorphism as permutation matrix would become : $\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$

Now let's come back to the case where we link the two boundaries neurons with one another.

The state matrix becomes :

$$A'(x) = \begin{pmatrix} 0 & -1 & 0 & 0 & -1 \\ -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 \\ -1 & 0 & 0 & -1 & 0 \end{pmatrix}$$

A' is $\Gamma - \textit{equivariant}$ where Γ contains the identity element and all the reflections of the pentagons with respect to each element (subgroup of $\mathbf{S_5}$). It is an isomorphism with the group containing the identity $(5 \cdot 5)$ matrix and the following ones :

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

One can easily observe that when we removed the two connections, the equivariant group became a subgroup of Γ. These 5 permutations matrices correspond to the 5 permutations with respect to each neuron as can be see on figure 32. It can be observed that the 4 of the matrices are just a circulant permutation of Θ and the other one is Θ.

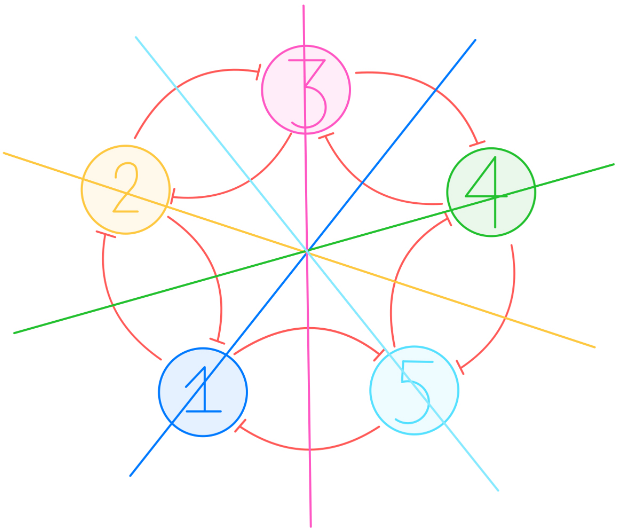


FIGURE 32 – Possible Symmetries on complete model

1.4.1 5 neurons C_2 equivariant model

We begin our cadence implementation using the $\mathbf{C_2} - equivariant$ circuit. We will use the same circuit implementation for both cases, because remember that one can use transmission gates as switches. In this situation, the control voltages for V_{15} and V_{51} are set to 0 while the other modulation voltages are all set to 1.8 Volts of course to ensure circuit proper connections.

One can simulate once more the model using I_{gain} as a bifurcation parameter. One decides to give a slightly asymmetry by imposing I_{in3} to 310 nA and the others to 300 nA. The results are shown on figure 34. One can observe that the solution is indeed symmetric with respect to the third neuron because the two directs neighbours of neuron 3 (in green), the 2$^{\text{nd}}$ and 4$^{\text{th}}$ are the most inhibited (in yellow), therefore they inhibit less strongly their direct neighbors neuron 1 and 5, so that's why these lasts have the same states (in purple). The isotropy subgroup of the solution after bifurcation becomes $\mathbf{S_1} \times \mathbf{S_2} \times \mathbf{S_2}$

Now, one interesting point to demonstrate that this model is not $\mathbf{S_5} - equivariant$ can be to simulate it in the same conditions, excepted that now we give the asymmetry to the second neuron. If the system was $\mathbf{S_5} - equivariant$, the curves of neuron 1 and 3 should this time be merged as well as curve 5 and 4. However, as you can observe on figure 35, this is definitely not the case as there is no connection between neuron 1 and 5, the second highest input is for neuron 5 because it's the furthest from neuron 2 in this configuration.

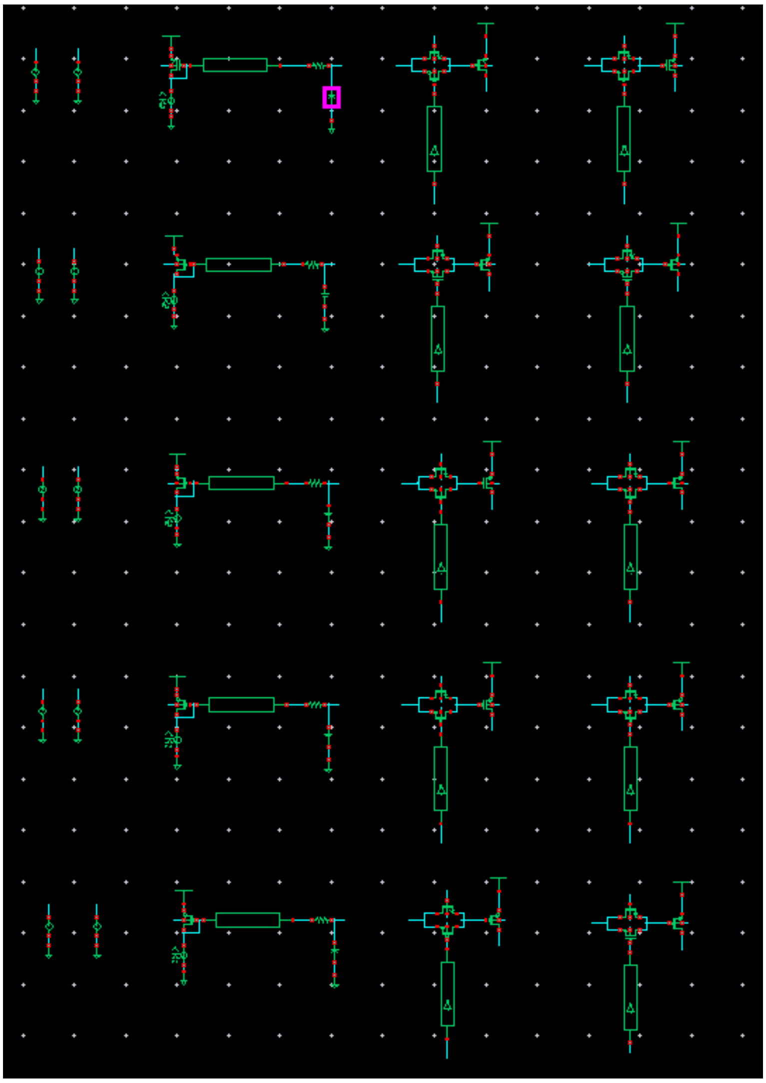


Figure 33 – 5 neurons circuit Implementation

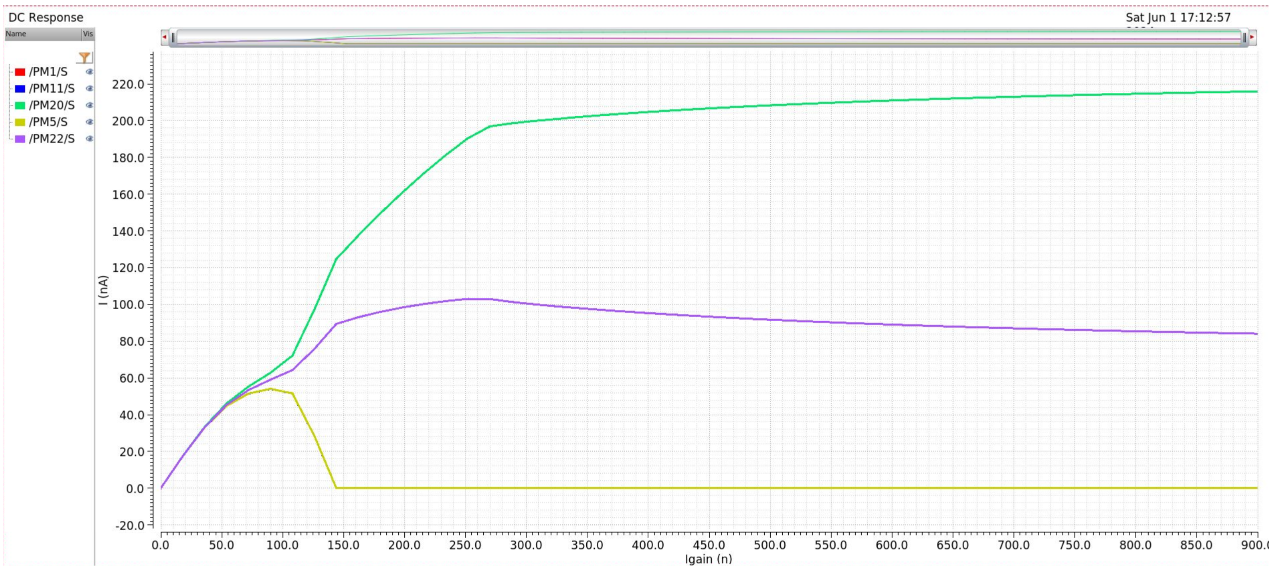


Figure 34 – 5 neurons bifurcation Diagram

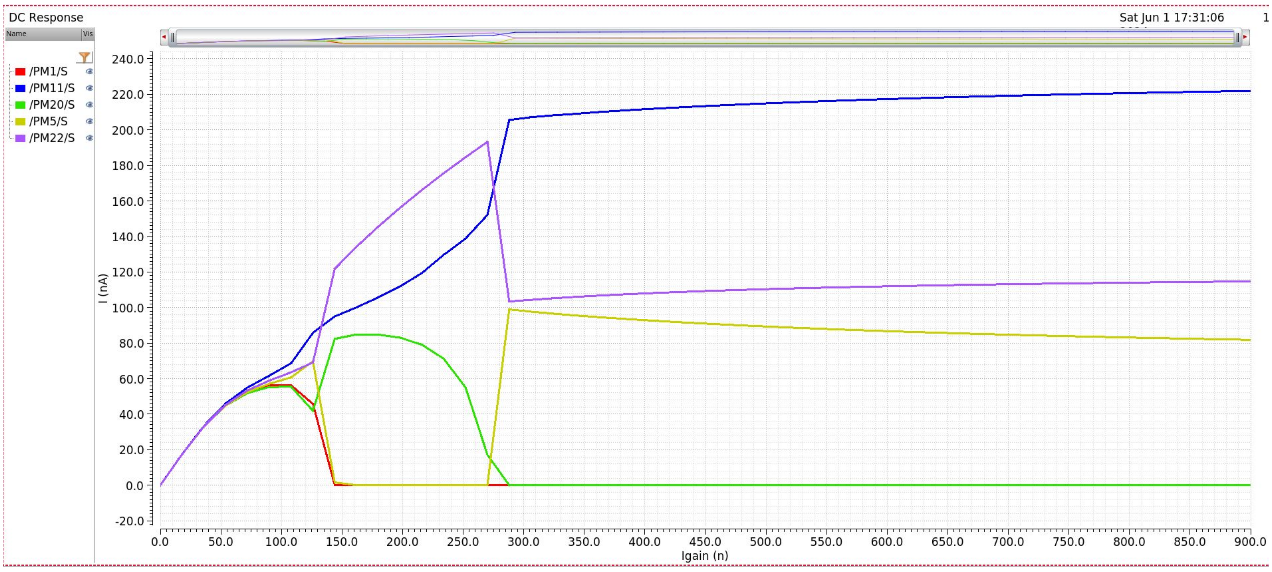


FIGURE 35 – 5 neurons bifurcation Diagram when neuron 2 has slightly higher input

1.4.2 5 neurons S_5 equivariant Model

The only change with respect to the previous circuit is that now $V_{15} = V_{51} = 1.8V$ so the circuit is perfectly symmetric.

Let us repeat the same simulation than what we have just done to see if there is any difference. In the figure 36, we set the second neuron input current at $310nA$ and the others at 300nA. One can deduce that the neuron 1 and 3 (in red) are the most inhibited (which is perfectly coherent with the fact the second is at the highest state) while the 4 and 5 are at the middle state (in purple) and finally the second is indeed at the highest state in blue.

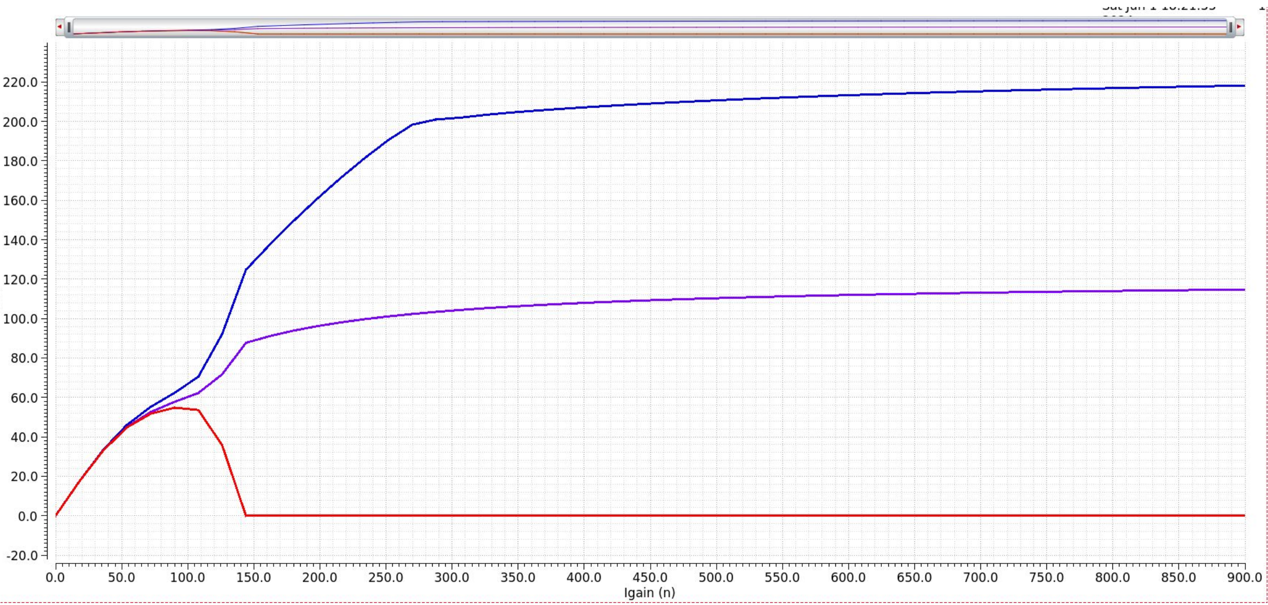


FIGURE 36 – 5 neurons bifurcation Diagram when neuron 2 has slightly higher input

We also achived to study other behaviors of the circuit exploiting this symmetry. For example, when we put slightly higher input currents for two neighbor neurons, both go to 0 (green curve) and the highest state neurons are the one neighbors of these (yellow curve), the most remote neuron (5th) is thus the most inhibited since it's the direct neighbor of highest state neurons (purple curve) (plot 37).

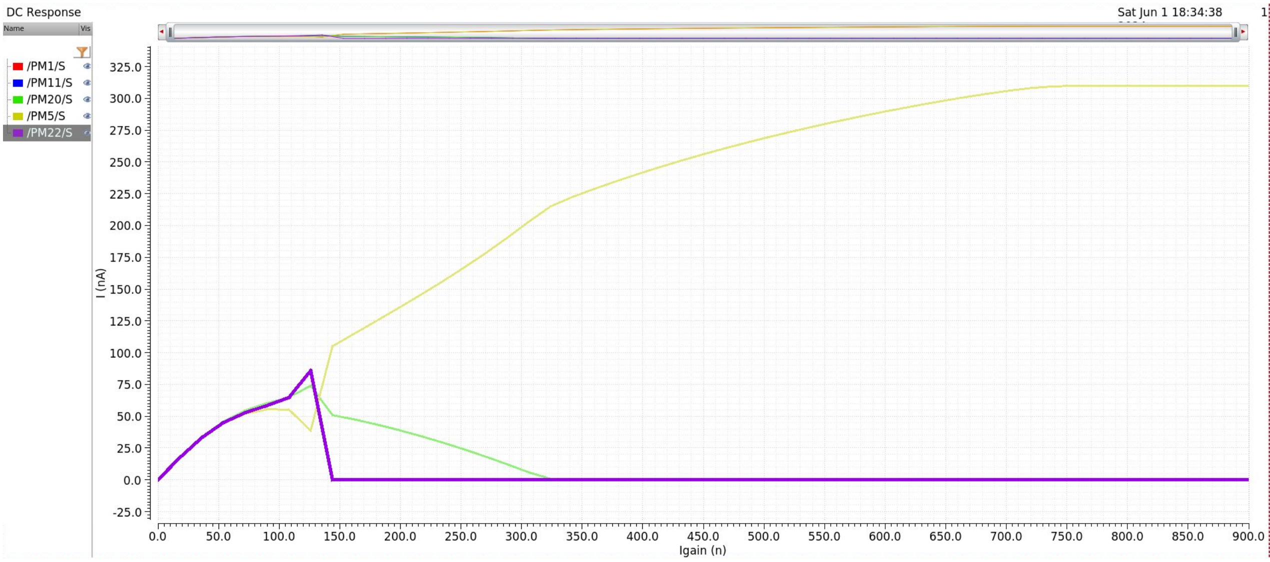


FIGURE 37 – 5 neurons bifurcation Diagram when neuron 2 and 3 have slightly higher input

We also performed several transient analysis that confirmed us that the bifurcation point was around 150nA. We also notice an amazing feature that didn't appear in DC simulations. The plot 38 is a transient analysis made in the same conditions than on plot 34. But what's very suprising in it, is that we see another bifurcation happening between the neuron 3 and 4 (equally spaced from neuron 1). One can observe that the DC simulation actually plots the value around 100ms, but transient analysis' solver allows deeper understanding of the ciruit. Here, the integrated numerical noise makes that neuron 4 eventually thrive over neuron 3, and that cannot be seen on the plot 34.

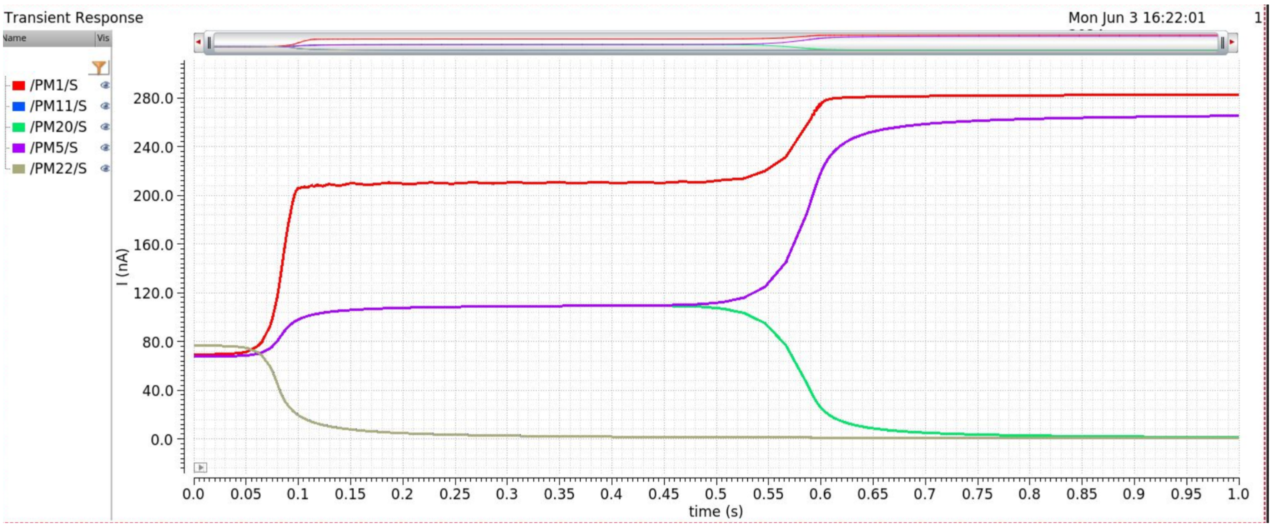


FIGURE 38 – 5 neurons transient simulation with input 1 slightly higher

Finally, one can briefly analyze the effect of the parameters of the sigmoid circuits to confirm what we said in the beginning of the project.

— Increasing I_{lin} will change the linear region of the sigmoid (increasing the linear range), the effect on the simulation is that the steady state of the simulation arise later.
— Increasing I_{thr} causes the bifurcation to arise earlier and the voltages are lower since we need to reach this threshold voltage to pass the zero response of the sigmoid.

2 Conclusion

In this project, we embarked on a comprehensive study of symmetry breaking in symmetric neuron networks, motivated by our desire to understand the intricate interactions between neurons. Our journey began with the detailed exploration of the sigmoid building block, which serves as a fundamental component in neuromorphic circuits. By thoroughly understanding the sigmoid function and its implementation, we laid a solid foundation for the subsequent phases of our research.

In the second phase, we utilized the Julia programming language to find the precise parameters at which bifurcation occurs in neuron networks. This computational approach allowed us to accurately identify critical points of symmetry breaking, providing valuable insights into the dynamic behavior of these networks.

The final and most extensive part of our project involved the implementation of neuron circuits in Cadence. Starting with a simple two-neurons circuit, we progressively scaled up to three and five-neurons configurations. Throughout this process, we meticulously explored the effects of symmetry on these circuits, observing how symmetry breaking influences their behavior and performance.

In conclusion, this project has significantly advanced our understanding of symmetric neuron networks and the role of symmetry breaking. By integrating theoretical exploration, computational methods, and practical implementation, we have gained a holistic view of neuromorphic signal processing, paving the way for future research and applications in this fascinating field.