CHAPTER IV

EXPERIMENTAL RESULTS

In this chapter we show experimental results of the design presented in the previous chapter and demonstrate that the implementation of the network we have presented can successfully segment images.

The first characteristic we have studied is the effects of biasing on mismatch. We show that differences in frequency between oscillators increase as biasing currents decrease and we quantify this variation.

After this, we describe the measuring system used to test the chip and the network response to segmentation problems. We show the ability of the circuit to segment images and analyze how it is affected by different biasing values of analog parts of the net.

Then, we show that the network is also robust against noise in the input image although it cannot eliminate it from the result. Finally, we study the time the network takes to segment images and the power it needs to do it. The chapter ends with some concluding remarks.

IV.1 FREE OSCILLATOR CHARACTERISTICS AND MISMATCH

Now, we show the results of oscillator measurements when oscillations are free, that is to say, no coupling synapses exist.

In chapter II, we have shown that an important element to consider when implementing such networks is mismatch. Synchrony is easily maintained when it does not exist or it is very small. However, this non-ideality can desynchronize oscillators if it is beyond certain ranges leading to malfunction of the network. As there is a trade-off between mismatch vs. power consumption and area of VLSI circuits, it is important to know which are the boundaries beyond which the network cannot work properly but also how each biasing parameter affects network performance.

First, we have performed several measurements of the period of each oscillator under two different biasing conditions (Figure IV.1). Higher biasing currents correspond to shorter periods that are represented at the bottom of the plot. A vertical line that comprises all different measures obtained represents each oscillator period. Longer lines represent larger differences between measures. Obviously, mismatch is reduced as biasing is increased but the plot also shows that measured periods are stable for the same oscillator during all the experiments. Vertical lines, which show differences between experiments, are quite shorter than differences between oscillators; thus, measures are not affected by noise, interferences and other operating conditions, at least one order of magnitude less than they are affected by device mismatch between oscillators.

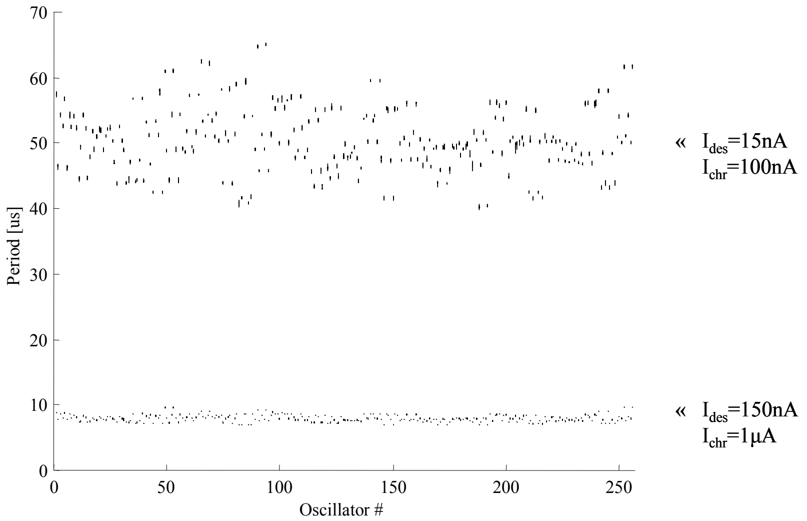


Figure IV.1: Period of each oscillator of the network under two different biasing conditions (top and bottom). Differences between oscillators (x-axis) are larger than differences between different experiments with the same oscillators (height of each vertical line)

In Table IV.1 we show the mean standard deviation of measures for different oscillators and for different experiments. Both values are related approximately by a factor of 8.5, from which we can conclude that accuracy of measures is much better than mismatch between oscillators.

	I_{des}=15nA; I_{chr}=100nA	I_{des}=150nA; I_{chr}=1µA
mean std between oscillators	4.8µs	544ns
mean std between experiments	200ns	24ns

Table IV.1: Mean standard deviation of periods for different oscillators and different experiments.

To quantify how characteristics change under different conditions, we have measured the period of each oscillator of a 16x16 network as a function of different biasing conditions. Results are summarized in Figure IV.2. Beyond values shown in that figure and for the specific biasing conditions used in these experiments, we have observed that cells stop oscillating.

This figure shows the ratio as a percentage between the standard deviation and mean value of periods of all oscillators as a function of different biasing values. From this figure, it follows that mismatch between oscillators does not have strong dependency on position or width currents. However, mismatch strongly depends on discharge current. It is logical consequence of working with such small currents where differences are more evident. Although not shown in the figure for clarity, charge current and reference voltage values are not important in terms of mismatch.

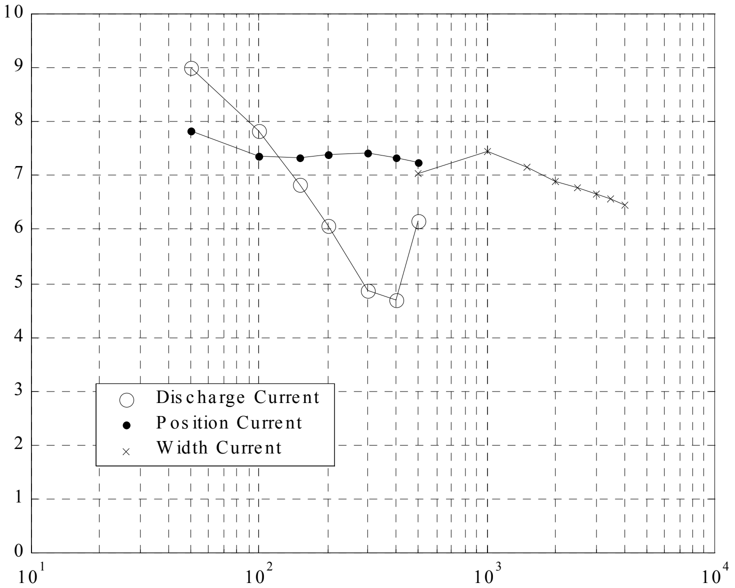


Figure IV.2: Ratio between standard deviation and mean value of period of 256 non-connected oscillators as a function of different biasing values.

I_{wid}	I_{pos}	I_{des}	I_{chr}	V_{ref}
1µA	300nA	120nA	1µA	1V

Table IV.2: Biasing conditions in Figure IV.2

Next we show some frequency plots of the free oscillators as a function of different biasing currents. The first plot (Figure IV.3) shows oscillator frequency as a function of I_{pos}. Frequency increases as bias current increases due to the shift of hysteresis cycle to higher thresholds. This effect combined with the quadratic relation of integrator capacitor voltage and input current to the comparator, reduces hysteresis width, hence, oscillation frequency is increased.

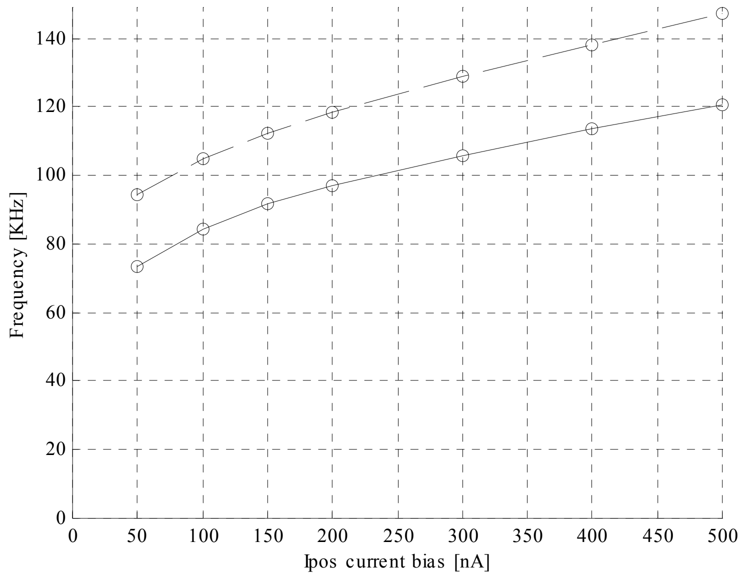


Figure IV.3: Oscillator frequency as a function of I_{pos} current biasing. Experimental results: solid line. Simulated results: dashed line

In Figure IV.4, we show oscillation frequency as a function of width current. This plot shows that frequency decreases due to the hyteresis width increase. However, this increase is non linear because of the quadratic relation between voltages and current thresholds.

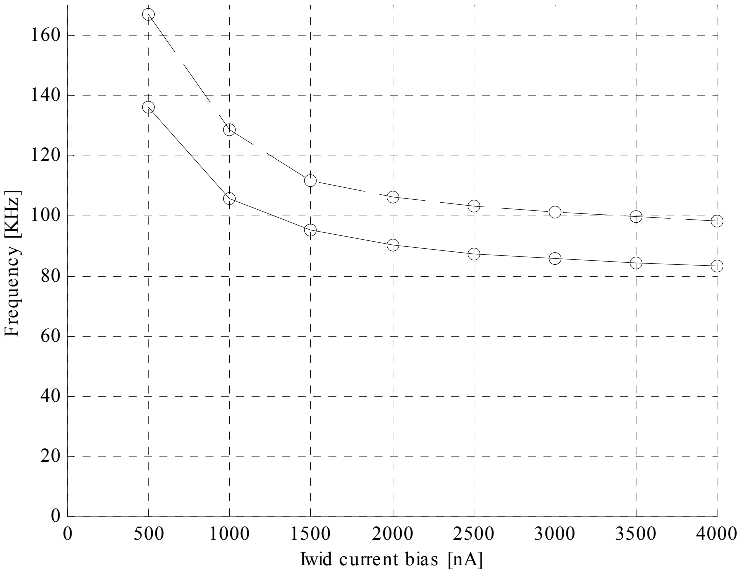


Figure IV.4: Oscillator frequency as a function of I$_{wid}$ current biasing. Experimental results: solid line. Simulated results: dashed line

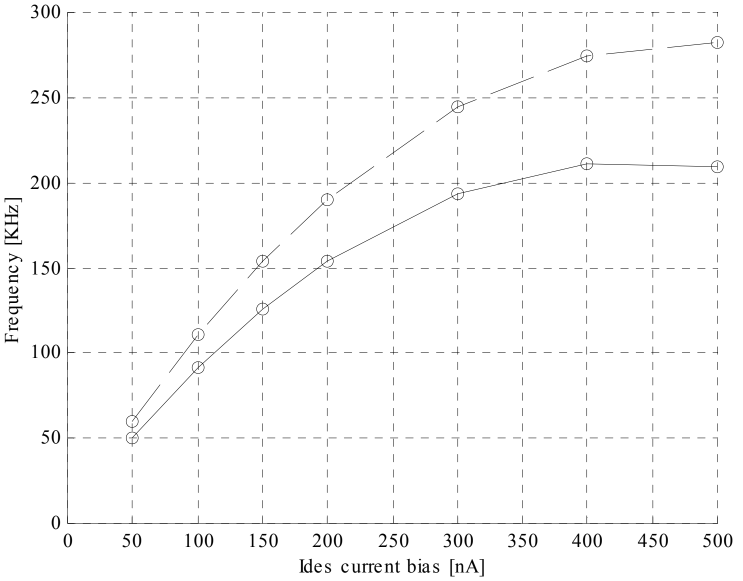


Figure IV.5: Oscillator frequency as a function of I$_{des}$ current biasing. Experimental results: solid line. Simulated results: dashed line

Frequency is mostly sensible to discharging current of the integrator than other bias currents because it is the only responsible of discharging integrator voltage during the oscillator silent phase. Figure IV.5 shows this property. Notice that the silent state is kept longer than the active state since network computing abilities depend on it. However, this frequency increase is limited by the duty cycle (active state/silent state ratio (η) defined in chapter II). As duty cycle increases, the assumption that frequency mainly depends on the discharging state is less precise and charging cycle must be considered. During this cycle, the capacitor is charged by I$_{chr}$-I$_{des}$, thus, active cycle length increases as I_{des} increases. The overall effect of both changes, reduces frequency rise as shown in the plot.

Finally, in Figure IV.6, we show that I_{chr} does not affect oscillation frequency when discharging current, I_{des}, is much smaller than it (I_{des}=120nA<<I_{chr})

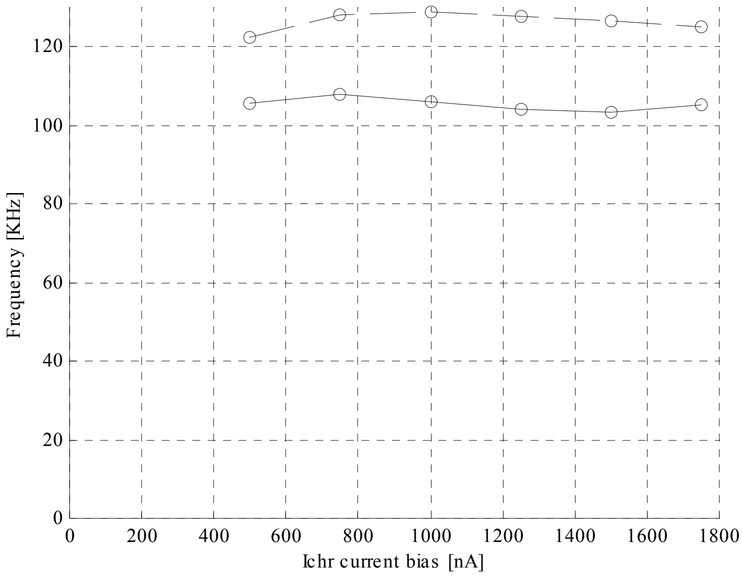


Figure IV.6: Oscillator frequency as a function of I_{chr} current biasing. Experimental results: solid line. Simulated results: dashed line

IV.2 EXCITATORY SYNAPSE CHARACTERISTICS

Another important biasing value is excitatory synapse current. This current is responsible of positively coupling oscillators to synchronize them. This current is normalized for each oscillator; thus, the sum of exciting currents is equal for all oscillators when they receive excitation from all their coupled neighbors regardless of their number.

This current is closely related to the strength of synchronization and above all, as it is responsible of charging the parasitic capacitance of the output node when it shifts to active, delays between oscillators strongly depend on this value as demonstrated in chapter II with the output delay oscillator. Thus, to reduce the width of the inhibitor pulses, and increase the number of different groups that can be placed in each cycle, delays should be reduced and excitatory synapses strength increased.

In Figure IV.8 we show this dependency between excitatory synapse strength and delay between oscillators. The plotted value is the mean delay between rising edges of oscillators of a 16 cell-line as a function of excitatory biasing. For lower values than shown in the plot there is no synchronization and for higher currents, oscillators reach a static state and do not oscillate.

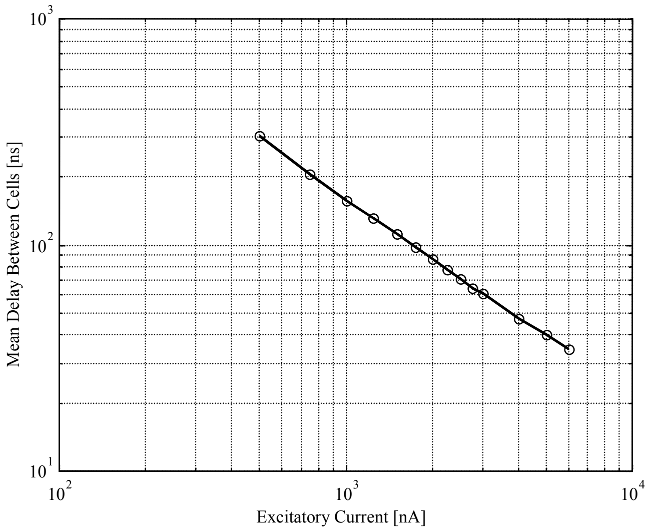


Figure IV.7: Mean delay between cells of a chain of coupled oscillators as a function of excitatory current.

IV.3 MEASUREMENT OUTPUT INTERFACE

As the network is a highly parallel system, it is not easy to continuously monitorize the evolution of all 256 cells. There would be necessary as many output lines as cells in the network, which would make the design not practical. Thus, for this design, only 16 output lines -a row at a time- are available. It makes necessary sampling each row of the network at different time steps. The sampling step depends on biasing of the activity detector of the network as seen in the previous chapter. This delay is on the order of some hundreds of nanoseconds or units of microseconds for each column, which makes a straight approach not feasible.

To overcome this problem we have decided to sample each timestep of the network at different periods, thus, it is necessary that frequency of each oscillator is stable enough to do it. The triggering signal is taken from a direct output of a lateral oscillator.

To detect activity of oscillators we read the state of the whole network by charging the output memory element of each oscillator by the 'load' signal as explained in the previous chapter. Then, the 256-bit information is led one row at a time to the 16-bit output where is captured by a digital acquisition card of a personal computer and presented to the operator or led to higher stages. Once finished that process, we have a binary 16x16x*n* array, where *n* is the number of samples. Each one of the 16x16 subarrays represents the network state at a certain timestep and each element of the subarray the state of each oscillator (1 for active oscillators, 0 for silent ones).

Note that outputs are buffered through inverters, thus, we cannot distinguish the exact form of output of oscillators but only their digital approximation: whether they are active (V_{dd}) or not (gnd).

IV.4 SEGMENTATION

In this section we present the result of applying the image at the top of Figure IV.8 to the network. The circuit has been biased as shown in Table IV.3.

Oscillator activity is shown at the bottom of Figure IV.8. Each image represents the array at a different timestep and each pixel represents the activity of its oscillator. It is black if the corresponding oscillator is active or white if it is in the silent state. Thus, looking at the network state at the right moment we can distinguish different objects that compose the image -a square (lower left), a cross (upper left), an arrow (right) and the background).

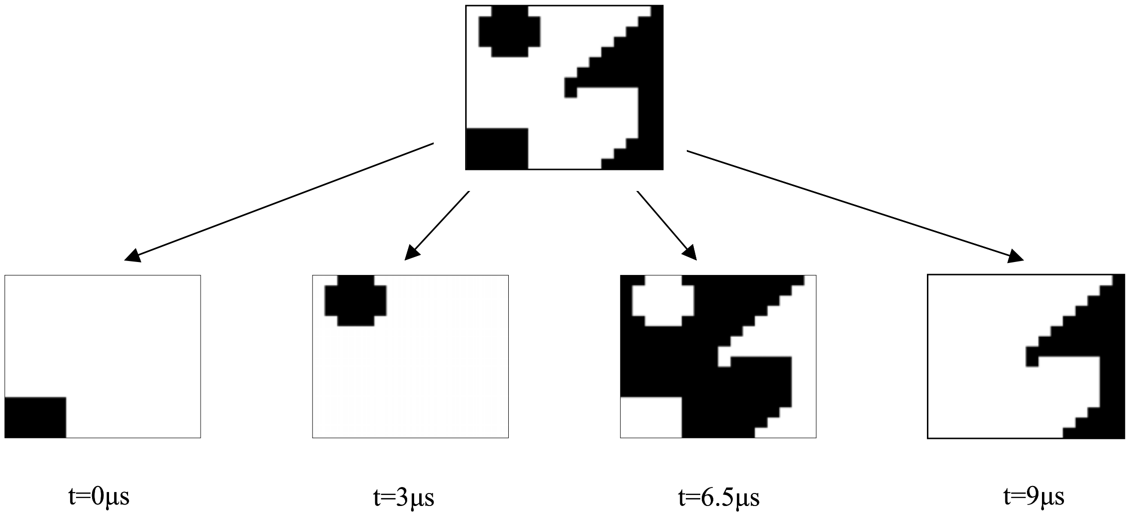


Figure IV.8: Input image mapped to oscillators (top) and network state at 4 different timesteps (bottom). Black pixels represent active oscillators and white pixels silent oscillators.

I_{pos}	I_{wid}	I_{des}	I_{chr}	I_{exc}	I_{inh}	V_{ref}
300nA	1µA	50nA	1µA	3µA	120nA	1V

Table IV.3: Biasing currents and voltages for Figure IV.8, Figure IV.9 and Figure IV.10.

In Figure IV.9 we show activity of the first row of the array. Signals belong to the cross (oscillators 2 to 4), to the arrow (oscillator 15) and to the background (the rest of oscillators). In can be easily seen that all oscillators have the same frequency and cells that belong to the same object oscillate in phase while groups of cells that belong to different objects reach their active state at different timesteps.

However, an important problem arises when looking at these results. It is how to choose the right timestep to sample the network state. A solution is found by means of the activity detector that activates the global inhibitor. This element can detect the activity of any pixel in the network; thus, its activity (Figure IV.10, bottom) shows the right timestep to sample. When this element is active, we know that the group of cells associated to one object is in its active state while others are silent.

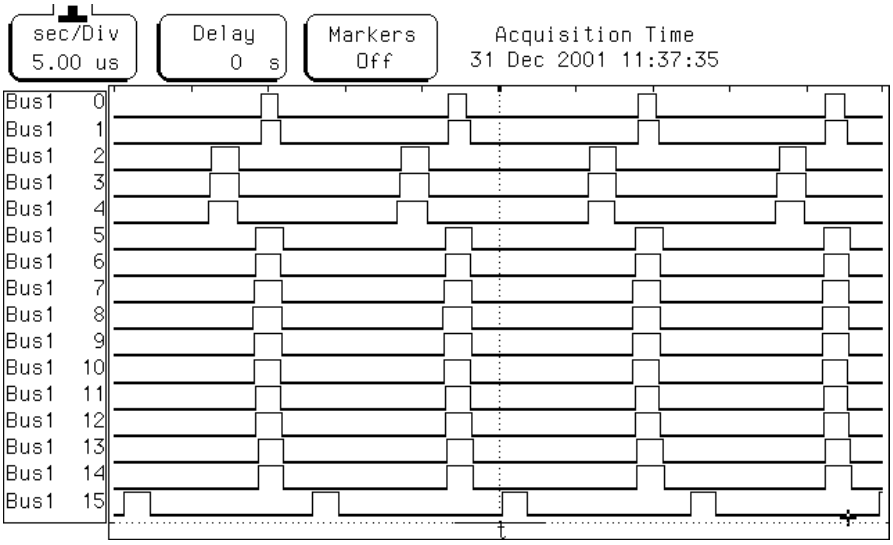


Figure IV.9: Activity of oscillators of the first row of the image. Pixels number 0, 1 and 5 to 14 are mapped to background. Pixels 2 to 4 are mapped to the cross and pixel 15 to the arrow.

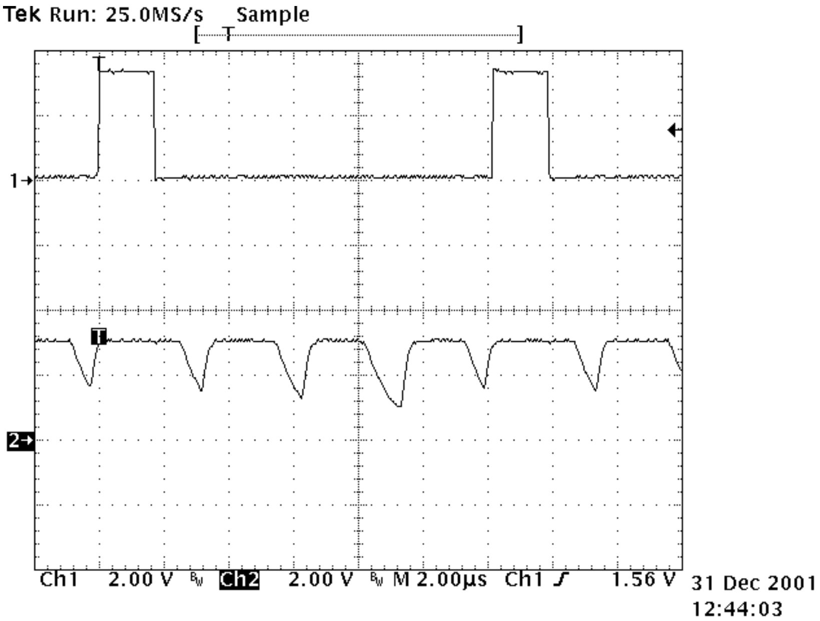


Figure IV.10: Background oscillator activity (channel 1, top) and activity detector response (channel 2, bottom) as seen in an oscilloscope. Note that four activity detector cycles occur at each basic oscillator cycle.

IV.5 DELAY EFFECTS

In chapter II, we have demonstrated that time delays due to parasitic capacitances at output nodes combined with mismatches are responsible of preventing the network to reach perfect synchrony. The result was that directly coupled oscillators that belong to the same object shift to active state with a slight delay. This delay is added for each oscillator that is excited and finally, two indirectly coupled and distant oscillators may have a delay even longer than the time they are active, thus activating the last one when the first one has gone back to silent (as oscillators #1 and #5 in Figure IV.11). However, it can be easily guaranteed that distant and highly delayed oscillators are indirectly coupled by checking the activity detector, because it is kept active during the active cycle of all coupled oscillators. When all oscillators of the same object are back to silent state, the activity detector also goes to silent until the next object appears. This fall and rise of the activity detector can be easily perceived and mark the end of one object and the beginning of the next one.

The effect of delays can be quantified by the ratio between the active time of the activity detector to the active time of a single oscillator. The bigger the ratio is, the larger delays are.

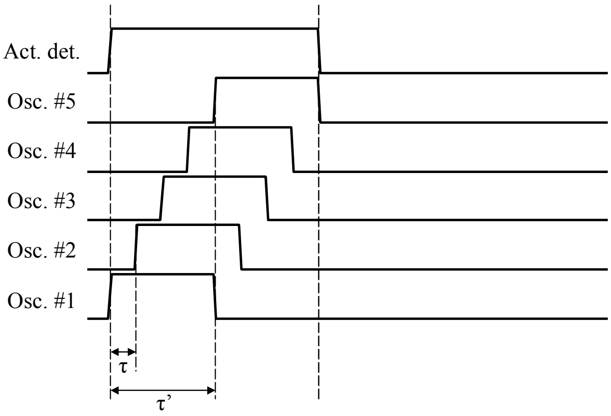
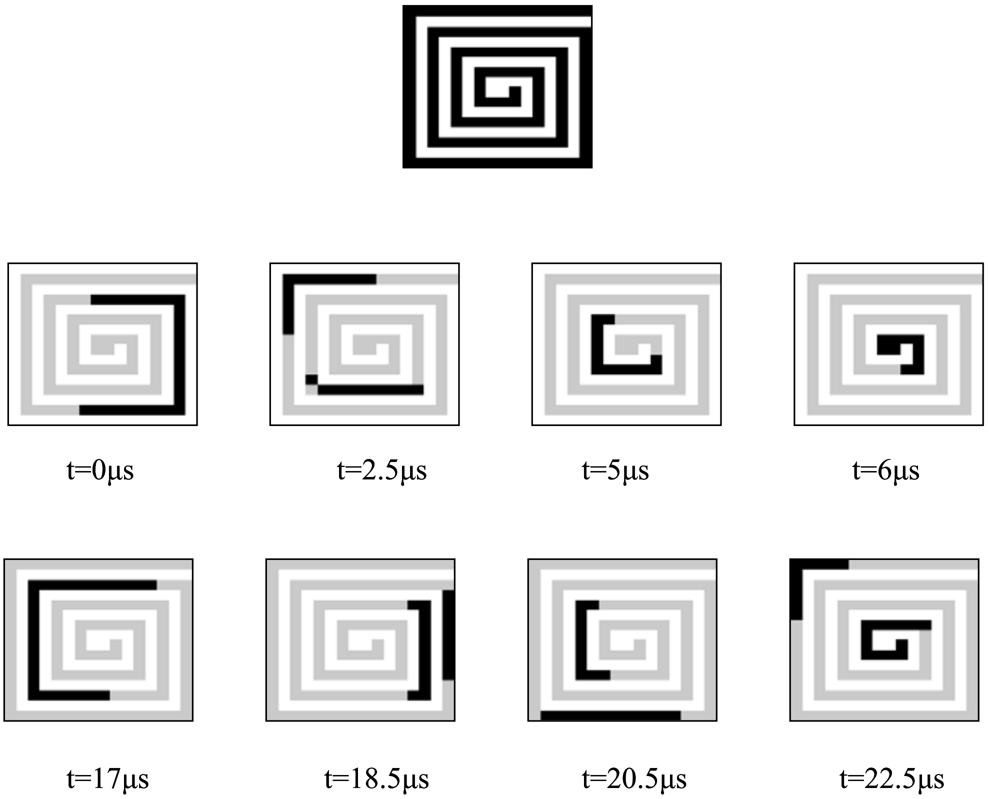


Figure IV.11: Delay of indirectly connected oscillators. Oscillator #1 and #2 are directly connected and their delay is τ, while delay between oscillators #1 and #5, which are not directly connected, is τ′ that is bigger than τ. The activity detector waveform is also shown at the top.

Next examples illustrate this case. A complex and non-common object in natural images as a spiral is used because it gives the longest path from two distant pixels in 2D array (Figure IV.12, top). For a better comparison with the previous example, biasing is the same as in Table IV.3 except that I_{des}=30nA, which makes oscillations slower and duty cycle smaller.



t=0μs t=2.5μs t=5μs t=6μs

t=17μs t=18.5μs t=20.5μs t=22.5μs

Figure IV.12: Input spiral (top) and network state at different timesteps (mid and lower rows). Black pixels represent active oscillators; white pixels and gray pixels, silent oscillators. Gray is used for clarity to indicate the object to which belong active pixels. Since only some timesteps are shown, some pixels do not appear as active in any plot although the network properly segments the image.

Figure IV.12 shows the input image at the top and activity of the network at different timesteps at the bottom. For clarity, we have used white and gray pixels to indicate silent oscillators. Difference between them lies in that gray pixels point oscillators that are indirectly coupled to active oscillators; thus, the active spiral can be distinguished easily in the figure.

In Figure IV.13 we show the evolution of a single oscillator in the top and the global inhibitor at the bottom. This figure clearly shows the effects of delays on the global inhibitior, wich is wider than the single oscillator. This difference means that oscillators that are associated to each spiral shift to active during a wide period of time in each cycle and that they cannot be active simultaneously.

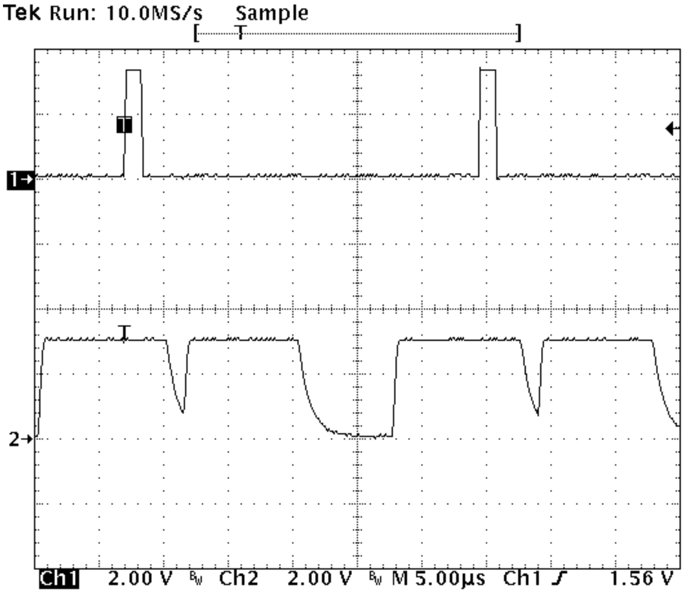


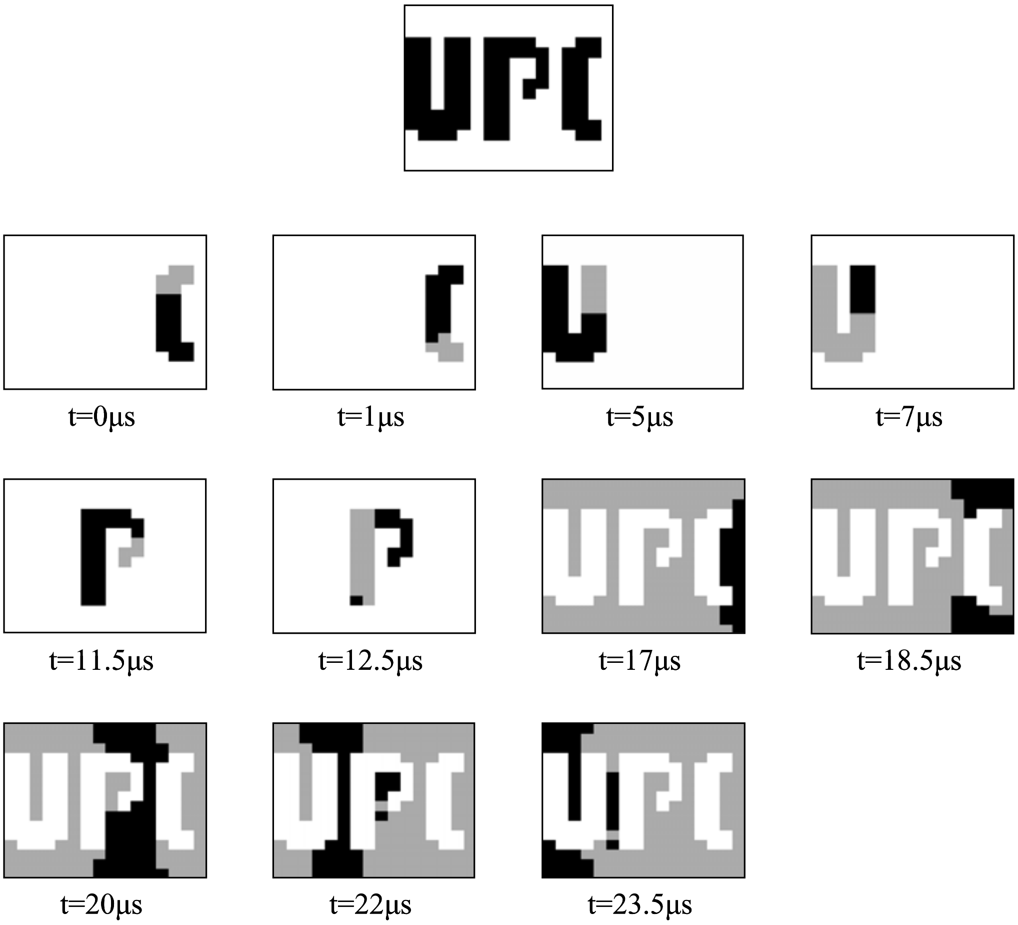
Figure IV.13: Oscillator (channel 1, top) and activity detector (channel 2, bottom) when input image is spiral in top of Figure IV.12.

However, a complex image as an spiral is not necessary to produce this effect. The next example in Figure IV.14 shows it. Image at the top of the figure is introduced in the netwok and its activity at different intervals is shown in lower lines. Gray and white pixels are used to indicate silent oscillators. As in the spiral example, not every oscillator that belongs to the same object is active simultaneously and some additional processing should be done to correctly distinguish each object, especially for the background, which is the bigger and more complex object of the image.

Bias values for this example are shown in Table IV.4.

I_{pos}	I_{wid}	I_{des}	I_{chr}	I_{exc}	I_{inh}	V_{ref}
115nA	500nA	50nA	500nA	3µA	250nA	1.36V

Table IV.4: Bias currents and voltages for image in Figure IV.14.



t=0µs t=1µs t=5µs t=7µs

t=11.5µs t=12.5µs t=17µs t=18.5µs

t=20µs t=22µs t=23.5µs

Figure IV.14: Input image (top) and network state at different timesteps (lower rows). Black pixels represent active oscillators; white pixels and gray pixels, silent oscillators. Gray is used for clarity to indicate the object to which active pixels belong.

In Figure IV.15 we show the activity of all 16 oscillators of row number 9 plus the inhibitor. Oscillators numbered from 0 to 4 belong to letter U. It is possible to find a timestep where all of them are active. We can say so of oscillators numbered 6 and 7, which belong to P, and pixels 12 and 13, which belong to C. However, oscillators that belong to the background would not be said to belong to the same object if the activity of the global inhibitor in line number 16 did not show it.

Although the circuit presented in this work is not able to group indirectly coupled pixels, it should be easy to be done in higher stages. If we look at the global inhibitor, it shifts to silent each time a whole group of oscillators of each object goes back to silent and before oscillators of another object become active. Thus, silent periods of the inhibitor indicate temporal boundaries between object activities. Detecting these shifts of the global inhibitor and grouping all oscillators that are active during each interval between shifts produces proper segmentation results.

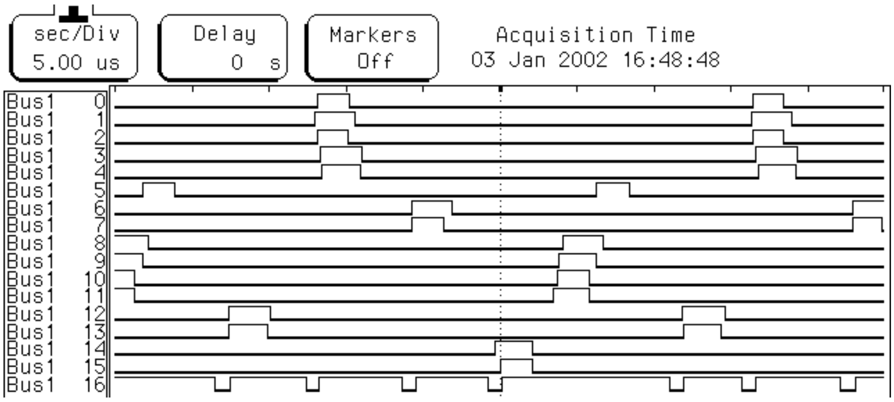


Figure IV.15: Activity of the 9th oscillator row (0..15) plus global inhibitor (16) for image in Figure IV.14.

IV.6 BIASING

A key element for the segmentation process to succeed is proper biasing of oscillators. In this section we show some examples of it. The input image used in this section is shown in Figure IV.16. It consists of two concentric black circles over a white background. However, as the inner part of the background is not connected to the outer part, the system must consider it as two different objects. Then, the image consists of four different elements.

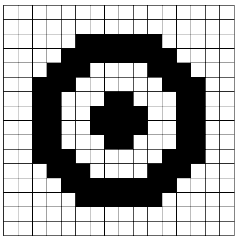


Figure IV.16: Input image

The first condition to take into account when biasing the network is the compromise between power consumption and mismatch. We want to reduce power consumption of the network but as mismatches increase as biasing currents decrease, the latter cannot be arbitrarily decreased. However, higher biasing currents do not always guarantee correct segmentation results as we show in next examples. Some limits exist in biasing currents for proper operation due to matching, power consumption and suitable limits of oscillation.

The main characteristic of oscillators to get proper segmentation is duty cycle. The smaller it is, the more active cycles can be placed in each period, thus, more objects can be segmented. In chapter III we demonstrated that charging and discharging currents are the best suited to modify this characteristic and next figures show it.

In Figure IV.17, we show activity of row number 8 of the network (lines 0 to 15) plus the global inhibitor (line 16). Note that oscillators of this row belong to all four objects, thus it is a good outline of array activity. Under these biasing conditions, all four objects (two circles plus two backgrounds) are perfectly segmented. Oscillators 6, 7, 8 and 9 belong to the inner black circle, oscillators 4, 5, 10 and 11 belong to the inner white background, oscillators 2, 3, 12 and 13 belong to the outer black circle and finally oscillators 0, 1, 14 and 15 belong to the outer white background. The global inhibitor at the bottom of the figure shows perfect grouping in spite of delays between oscillators, especially in outer objects –cells 0 and 1 vs. cells 14 and 15-, which are the bigger and more complicated.

In Figure IV.18, we have increased discharging current, which increases oscillation frequency plus duty cycle. However, four objects can still be segmented as shown. Note that the order in which oscillators become active differs from the previous example. When the network suffers any change in biasing or in the input image, the system evolves to a different state where the sequence of active objects may change. However, once it has been fixed, it is maintained until the next jump provided that biasing is correct.

In the next example, Figure IV.19, we have highly increased discharging current, up to 250nA. Now, duty cycle is too big as to allow the existence of four different objects, only two are possible. The result is that different objects are confused and not segmented –inner circle and outer background are grouped together and outer black circle and inner background are not distinguished.

Possible biasing currents extend to a great range, and they can be highly reduced. In Figure IV.20, we show it. Discharging current has been decreased down to 15nA and duty cycle has become very small but the network still works properly. More objects could have been placed in silent periods of the global inhibitor when the whole network is silent.

I_{chr} is the other biasing variable that is directly related to the duty cycle of oscillators. Effects of reducing or increasing this value are the same as modifying discharge current. However, even though the important key to possible number of segmented objects is the ratio between charging and discharging currents, matching gives a valid range for them. Their values cannot be too small and neither their ratio too high.

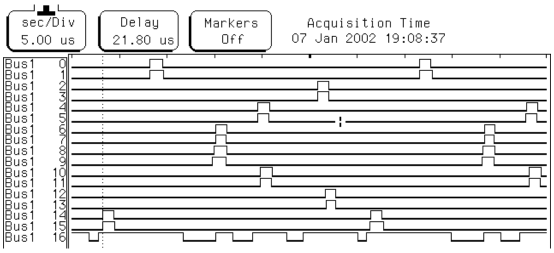


Figure IV.17: Activity of the middle row when I_{des} is 45nA.

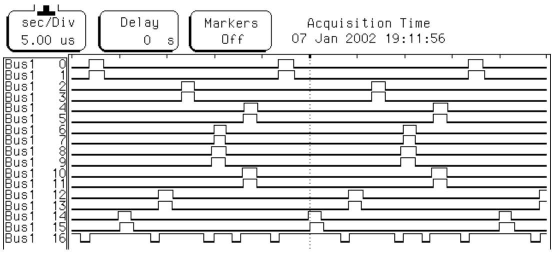


Figure IV.18: Activity of the middle row for a greater I_{des} (75nA)

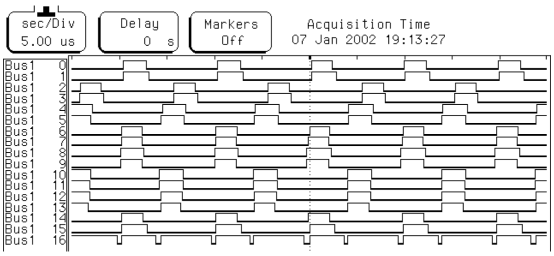


Figure IV.19: Activity of the middle row for I_{des}=250nA

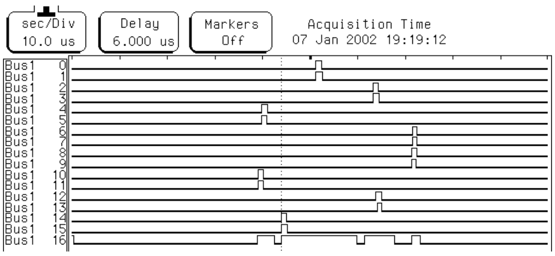


Figure IV.20: Activity of the middle row when I_{des} is very small (15nA). To check frequency, observe at the lower left corner inhibitor activity of the previous period.

Another critical value is inhibition current. This current is responsible of separating uncoupled blocks of pixels and prevents random synchronization between them. In addition to this, it is responsible of matching frequencies of blocks, thus, all cells oscillate with the same frequency.

In Figure IV.21 and Figure IV.22 we show two opposite cases. The first one shows a network with too low inhibition biasing. This inhibition current is not able to keep blocks segregated but it does not affect synchrony between blocks. Coupled oscillators are kept together, thus, objects are not divided, but random synchronization joints different objects during different periods, leading the system to wrong results.

The next plot, Figure IV.22, shows the activity for high biasing inhibition. There is also random synchronization between non-coupled oscillators as in the previous example but in this case some coupled cells are desynchronized. The system has evolved to some kind of behavior where not every oscillation can be predicted. Each oscillator can excite the global inhibitor and then inhibit itself and its neighbors to a degree that it stops all their oscillations. Only groups of synchronous coupled oscillators can oscillate. Note that in both cases, frequencies of oscillators differ; even in the latter case, periods of the same oscillator are not constant through time.

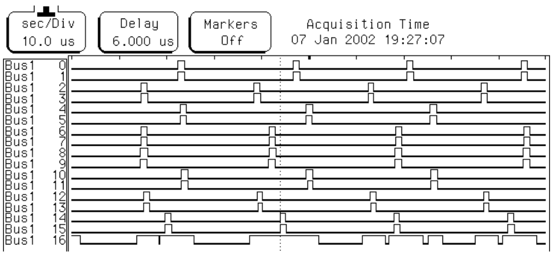


Figure IV.21: Activity of the middle row for low inhibition biasing (I_{inh}=20nA)

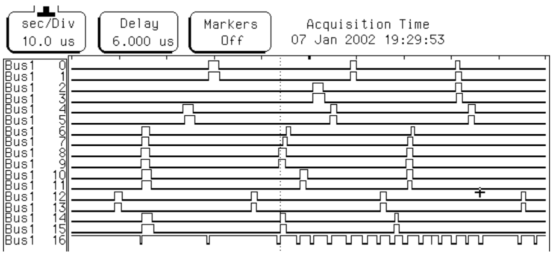
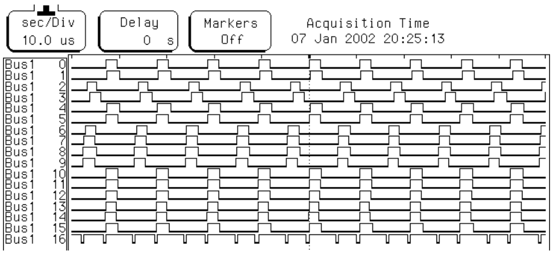


Figure IV.22: Activity of the middle row for high inhibition biasing (I_{inh}=820nA)

Next figures show that excitatory currents should be high to get good results. When these currents are not high enough, oscillatory frequency is too high and silent to active shift is too slow due to low currents that charge output capacitance of the comparator. Figure IV.23 shows network behavior when excitatory current is 500nA. From line 16 activity it follows that only two objects are distinguished. The inner black circle and part of the outer black circle (lines 2 and 3) are grouped together while other objects are mistaken as being only one object.

The other plot, Figure IV.24, shows the opposite case when excitatory current is high (I_{exc}=3500nA). Hysteresis cycle of oscillators has increased, thus oscillations are slower and shifts from silent to active states are faster, thus, more objects can be placed in one oscillation cycle. The main drawback of using such high currents is power consumption that increases as biasing current increases. Moreover, in our design, when excitatory current reaches 6500nA, oscillations stop and the system becomes useless.



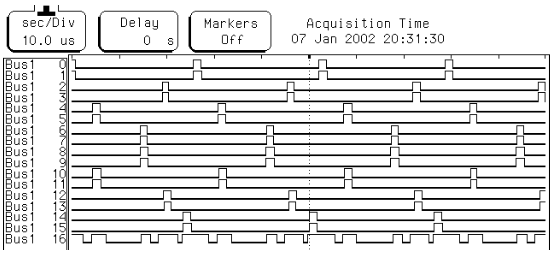
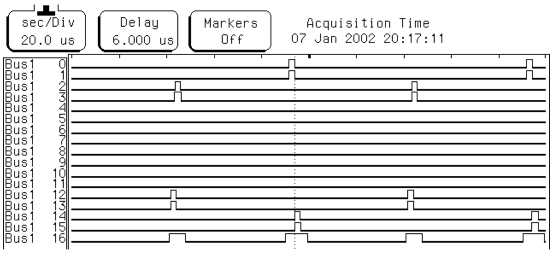


Figure IV.23: Activity of the middle row for low excitatory currents (I_{exc}=500nA)

Figure IV.24: Activity of the middle row for high excitatory currents (I_{exc}=3.5μA)

Position current is responsible of biasing cells to oscillate. If this current did not exist or it was too low as in Figure IV.25 (I_{pos}=1nA), some oscillators would not oscillate (cells 4 to 11). This biasing value can be increased as necessary if we assume that an increase in this value means a slight reduction of duty cycle, thus, less objects can be segmented at a time, as demonstrated in previous chapter. In Figure IV.26, we show the response of the network when this value is quite high and objects are not properly segmented. This value can be used to establish some kind of image noise immunity as presented below.



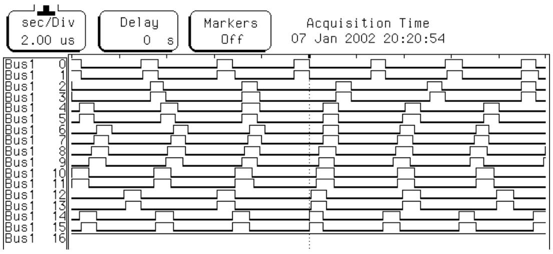


Figure IV.25: Activity of the network for very small position biasing (I_{pos}=1nA)

Figure IV.26: Activity of the network for high position biasing (I_{pos}=2500nA)

Finally, we do not have observed any noticeable effect when width of hysteresis is modified except slight differences in frequency and duty cycles as demonstrated in the previous chapter.

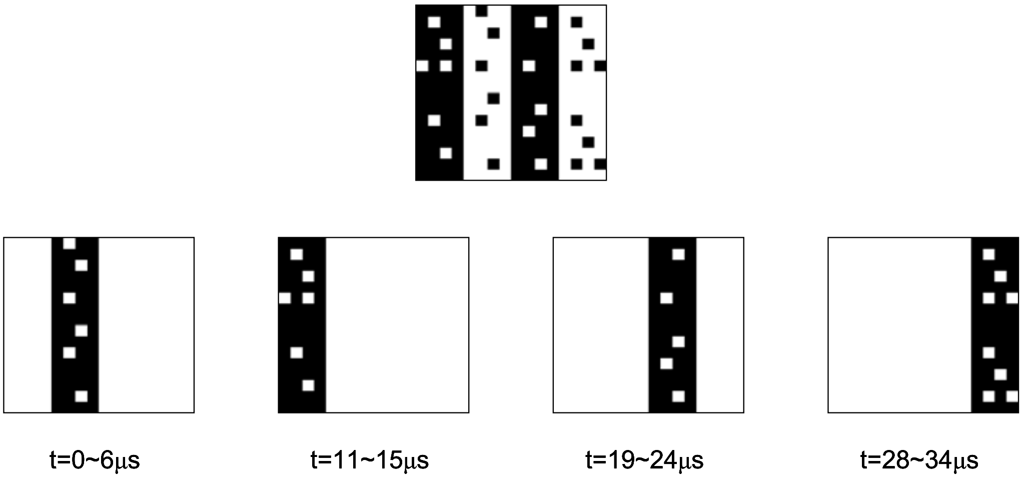
IV.7 IMAGE NOISE

An important aspect to be considered in image processing is that images are usually noisy and any algorithm must be robust against this effect. Particularly, in our algorithm, with a fixed biasing, the number of different objects that can be segmented is limited to the inverse of the duty cycle. If the number of objects is above segmentation capacity, different objects are grouped as being the same. If noisy pixels are segmented as objects, segmentation capacity is exceeded and the system fails. Thus, it is needed that only big groups of pixels are distinguished. Small groups of pixels or isolated pixels, as noise, should never shift to active or do it synchronously with a real object to not to occupy a temporal slot in the oscillatory period, thus, reducing the number of real objects that can be distinguished.

That problem has been solved by Wang and Terman [Wang and Terman'97] using the lateral potential. An extra term in oscillator equation that allows only oscillators that belong to big objects to jump to active without synapse excitation. Thus, oscillators that do not belong to big objects, i.e. noisy pixels not connected to other pixels or pixels with few connections, only shift to active when they are excited by a coupled oscillator. The overall effect of this term is that small objects cannot become active by themselves.

In our design, this effect is achieved by reducing the position biasing current (I_{pos}) to a small value. It must be small enough as to stop oscillations of isolated pixels in conjunction with inhibition current. Note that isolated pixels do not have any excitatory connection and this term is the responsible of allowing oscillations when I_{pos} is small.

In Figure IV.27, we show an example of this effect when a small I_{pos} is used (50nA). Image at the top of the figure is the input. It consists of two pairs of vertical bars with noise. The four plots at the bottom represent oscillators that are active at some time of different temporal slots. Only pixels that belong to big objects are able to oscillate, while isolated noisy pixels stay silent and never shift to active. Figure IV.28 shows the activity of row number 5 and the global inhibitor (line 16) during some periods. Oscillators associated with the same bar oscillate synchronous while oscillators associated with noisy pixels (0, 2, 5, 9, 13 and 15) never shift to active.


t=0~6μs

t=11~15μs

t=19~24μs

t=28~34μs

Figure IV.27: Input noisy image (top) and active oscillators (bottom) during four different temporal slots in each period.

In Figure IV.29 we have changed biasing current I_{pos} to 150nA. The increase of this biasing current has produced that oscillator number 5 oscillates, however, it is not able to occupy a temporal slot by himself and it is 'attached' to the third bar, not disturbing the correct behavior of the network.

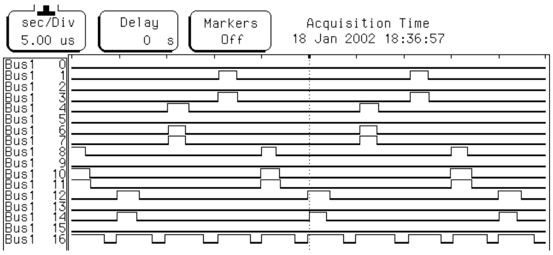


Figure IV.28: Activity of row number 5 for input image of Figure IV.27 and I_{pos}=50nA.

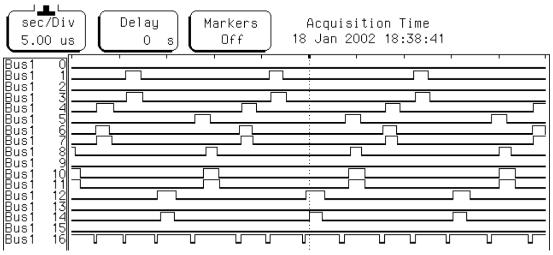
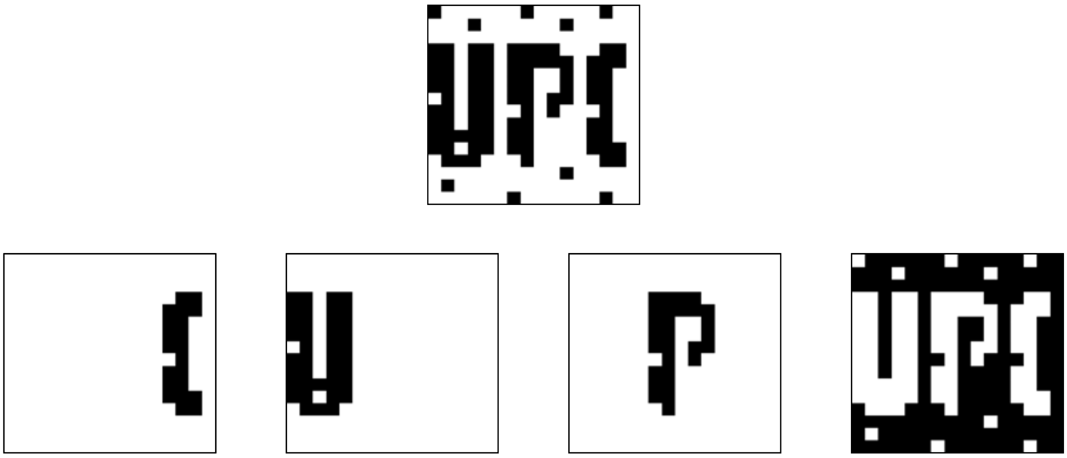


Figure IV.29: Activity of row number 5 for input image of Figure IV.27 and I_{pos}=150nA

In Figure IV.30 and Figure IV.31, we show another segmentation result of a more complex noisy image. The input image is shown at the top and oscillator activity during four different temporal slots of a period at the bottom of Figure IV.30. In Figure IV.31 we show the activity of a background oscillator plus the inhibitor behavior as seen in the oscilloscope. Note that the global inhibitor detects only four different objects because cells associated to noisy pixels do not oscillate.



t=0~2.5μs t=5.5~9.5μs t=12~16.5μs t=19.5~25.5μs

Figure IV.30: Noisy input image (top) and network state during four different temporal slots during one oscillator period.

These results show that the network is robust against noise although it cannot reconstruct the original noiseless image. This is a task for higher processing stages.

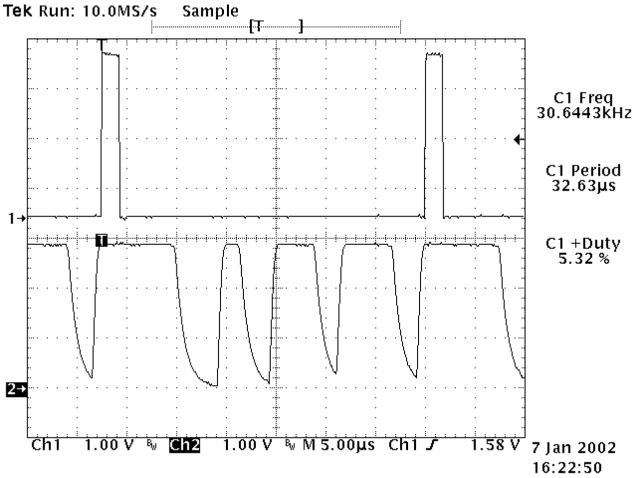


Figure IV.31: Background oscillator activity (top) and inhibitor activity (bottom) for input image in Figure IV.30. The length of each inhibitor pulse displays complexity of each object. The shortest belongs to letter 'C' and the widest to the background.

IV.8 SEGMENTATION SPEED

Segmentation speed mainly depends on strength of excitatory connections. Synchrony in astable oscillators is different from Fast Threshold Modulation in the sense that beyond a certain range of initial conditions, synchronization is not possible. This is not an important problem in electronic implementation due to oscillator matching because, as stated in chapter II, mismatch is responsible for taking oscillators close enough to synchronize if excitatory connections are too weak and they also produce some delay between coupled oscillators.

However, when big groups of oscillators must be synchronized, excitatory coupling must increase to keep synchrony. Thus, for practical applications, excitatory coupling cannot be weak. It means that power consumption increases as this value becomes stronger but also segmentation speed is increased.

In Figure IV.32 we show how two coupled oscillators synchronize under strong coupling (I_{exc}=I_{wid}=500nA). Coupling is established at triggering signal indicated by a *T* in the figure. It shows that synchrony is achieved in the next oscillatory cycle when both oscillators shift to active at the same time.

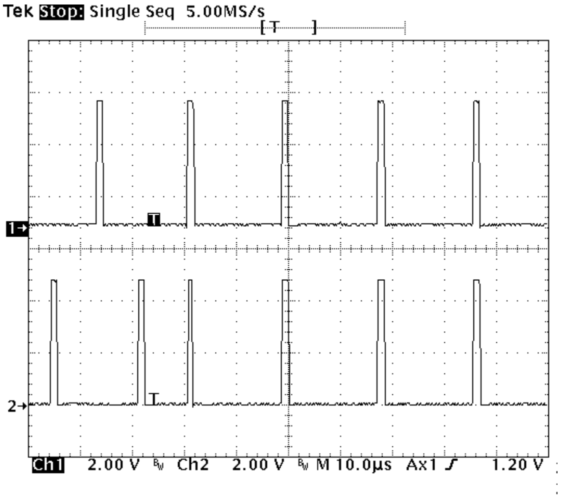


Figure IV.32: Synchronization of two coupled oscillators. Excitatory coupling (I_{exc}=500nA) is established at triggering signal (indicated by letter *T* in the image) and synchrony is rapidly achieved after one cycle.

In next figures (Figure IV.33) we show the same process under very weak coupling (I_{exc}=50nA). It demonstrates that synchrony is also possible under those adverse conditions even though it is much slower. Figure IV.33 shows the activity of both oscillators as seen in the oscilloscope. As the complete synchronization process could not be shown in one oscilloscope display, we show two consecutive displays.

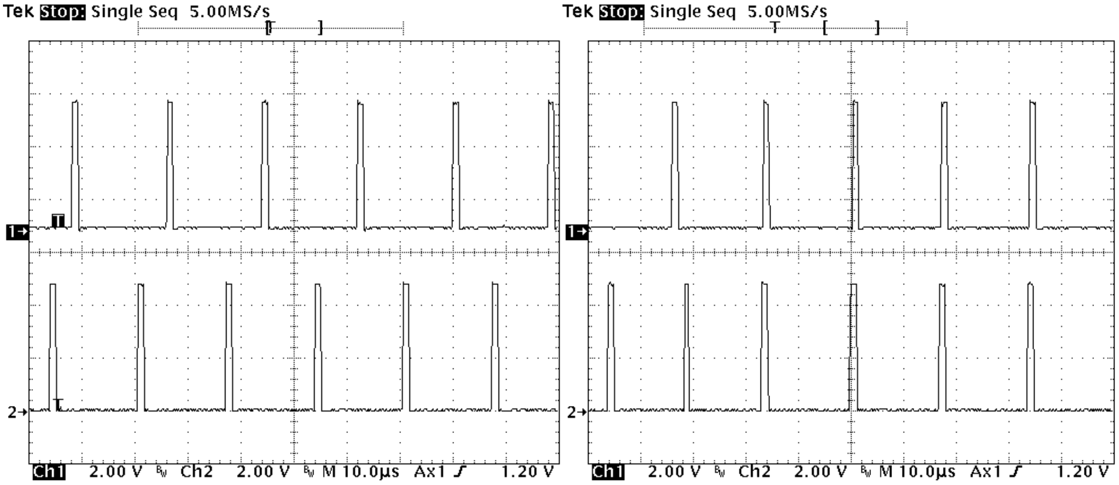
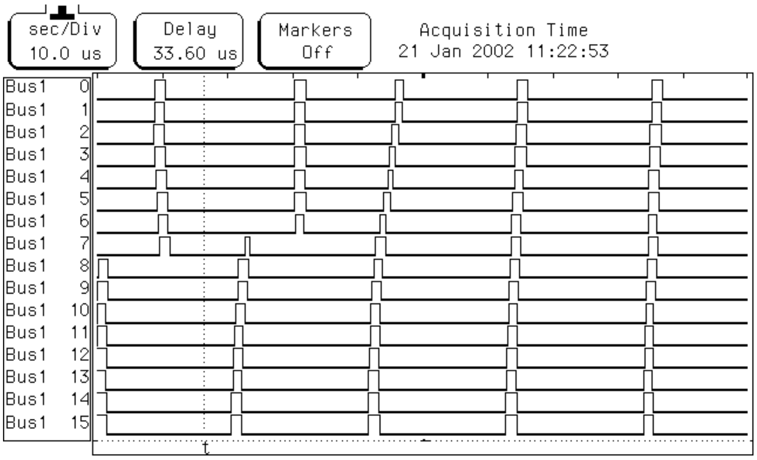


Figure IV.33: Synchronization of two coupled oscillators. Mismatch forces slow synchronization. Excitatory coupling is weak (I_{exc}=50nA) and established at triggering signal indicated by *T*. Two oscilloscope images with different delay are shown to appreciate the whole synchronization process.

Blocks of oscillators are more difficult to synchronize than isolated couples of cells. The next figure (Figure IV.34) illustrates how two coupled lines of eight oscillators synchronize when oscillators at their common boundaries are coupled together forming a 16-oscillator line.

Synchrony is slower than in the previous example under the same biasing conditions, however it is still fast and it is achieved in the second cycle after coupling of boundary oscillators, indicated by a vertical dotted line and a '*t*'.



sec/Div 10.0 us

Delay 33.60 us

Markers Off

Acquisition Time 21 Jan 2002 11:22:53

Figure IV.34: Synchronization of a line of oscillators. Before triggering, indicated by a *t* and a vertical dotted line after the first period, there are two uncoupled lines of eight oscillators. After triggering, a 500nA excitatory current couples both lines and synchrony is achieved after two cycles.

IV.9 POWER CONSUMPTION

Power consumption of the network has been tested under different biasing conditions and measurements for two different conditions that lead to correct results are shown in Table IV.5. For a 3.3V power supply, measurements show that the network can properly segment images at lower power consumption figures than 6µW/pixel.

	I_{wid}	I_{pos}	I_{chr}	I_{des}	I_{exc}	I_{inh}	V_{ref}	V_{ref_A}	Total Current	Total Power	Power/ Cell	Freq.
1	.1 µA	50nA	.25 µA	14nA	.5 µA	43nA	.8V	.2V	450µA	1.5mW	5.8µW	14KHz
2	1µA	150nA	1µA	60nA	3 µA	200nA	.8V	.2V	1.5mA	5mW	19µW	52KHz

Table IV.5: Biasing currents and corresponding power consumption and operating frequency for 3.3V power supply.

IV.10 PERFORMANCE COMPARISON

Comparing a testchip with other designs is not an easy task, especially if the testchip performs only a stage of a complex processing chain, as the one presented in this dissertation.

A key element when implementing an algorithm is the platform to use. Software implementations are easy to program and flexible if they have to be modified. However, they are slow and require large computing resources. These characteristics make them very well suited to investigate new algorithms and their properties. On the other hand, hardware implementations are more difficult to build and modifications are usually complex and not always possible. This makes them not appropriate for applications that must be constantly redesigned but they are very well suited when they have been successfully tested and they are to be used in final designs. This approach will allow us to exploit full possibilities that technology offers.

Comparing a hardware implementation with its software counterpart in terms of performance is tricky because each one has its own applications and the latter is not specifically intended to improve speed or power consumption. In addition to this, the first approach is parallel and the second one serial, thus, finding figures to be compared is not obvious. However, we give some figures to emphasize the advantages of a hardware implementation.

LEGION algorithm has been simulated on a HP 735 workstation [Linsay and Wang,98] using a standard 4th order Runge-Kutta method and another method especially developed for LEGION algorithm (known as Singular Limit). Results are given for a 50x50 pixel binary input image and figures are shown in temporal units per pixel to facilitate comparison in Table IV.6. In this table, we also show results for our design provided segmentation is achieved in two cycles, as demonstrated in this chapter. Figures show that the hardware implementation is much faster but this result is even more advantageous when power and size are considered. Obviously, a 100MHz workstation CPU is much larger, power eager and more expensive than a 6mm^2 chip at 14KHz, do not say when the analog network is compared to the whole workstation.

		Total segmentation time	Segmentation time/cell
LEGION	Runge-Kutta	1102 s (50x50)	441000 µs
	Singular Limit	10~4.5 s (50x50)	4000~1800 µs
G-LEGION		1.62~0.57 s (256x256)	24.7~8.7 µs
Analog VLSI		142~38 µs (16x16)	0.56~0.14 µs

Table IV.6: Execution time of various oscillatory segmentation network implementations

G-LEGION, an improved algorithm based on LEGION to simplify its complexity and computer load has been developed by Shareef et al. [Shareef et al.,99]. Performance of this algorithm in computerized tomography and magnetic resonance imaging is given in the same paper. This algorithm is not strictly an oscillatory algorithm although it is inspired by LEGION, thus, no differential equations are to be solved and performance is enhanced. It is important to stress that this algorithm performance is comparable to other segmentation schemes. Some figures in terms of segmentation speed are given in Table IV.6. Figures show that the hardware approach is 50 times faster per pixel than G-LEGION approach. It should be stressed that applications of both algorithms are quite different, thus, G-LEGION segments gray-level images instead of black and white ones and uses a wider excitatory coupling neighborhood of eight elements instead of four. However, these results are taken from simulations run on a high-end Silicon Graphics Onyx workstation, whose power consumption and volume are thousands of times larger than our implementation. Furthermore, our design is implemented using 0.8µm CMOS technology while computers used to simulate algorithms compared above are built on smaller, thus better, technologies.

To compare the circuit with an integrated low-power CPU, we have evaluated a common microprocessor core as ARM7TDI implemented on a 0.18µm process that consumes 0.25mW/MHz according to vendor data. Thus, from Table IV.5, and provided that our desing achieves a correct segmentation in two cycles, in terms of power consumption our design is equivalent to an ARM7TDI processor at 6MHz (the analog network consumption is 1.5mW and the ARM power consumption is 0.25mW/MHz), that is to say, it performs 771 instructions (0.9MIPS/MHz) in 142µs (two analog cell periods). Algorithms to segment and label binary images involve scanning the image by rows and forming equivalence classes. These equivalence classes are subsequently merged and the image's components labeled accordingly. This is a recursive algorithm and requires a large number of memory elements not computed in ARM microprocessor core figures shown above. Thus, less than 771 RISC instructions seem not enough to perform such an algorithm for a binary 256-element image. This difference in performance may increase in future designs if gray level images are used because the analog hardware implementation complexity should not considerably increase as the digital one will.

Referring to area, ARM's processor mentioned above is 0.53mm^2, that is to say, six times smaller than our design, however, it should be noticed that ARM processor is implemented using a 0.18 technology which compared to our 0.8 technology is about 20 times smaller (4.44^2).

For a rough estimation of our circuit complexity, it can be compared to a standard digital implementation using the same technology. In these figures, the D flip-flop is considered as a part of the oscillator, thus, the hardware implementation could incorporate a light sensor to input images instead of the shift register without increasing area. The total core area of our circuit is 2,95mm^2 and it consists of 256 cells. Thus, 11.523µm^2 are needed for each cell. Two-input NAND standard cells occupy 416µm^2. If wiring for each cell occupies the same area, this gives a result of 800µm^2 for each gate and the cell area is equivalent to approximately 14 gates or 56 MOS transistors. Power consumption of a 0.8µm standard two-input NAND gate is 1.73µW/KHz, thus, power consumption of each oscillator is equivalent to power consumption of 14 gates at 240KHz.

Finally, a comparison with a specific digital VLSI parallel implementation should be analyzed in detail. As a first estimation, we can state that 3 bits per cell are needed to label segments and excitatory synapses are to be implemented in addition to some external control circuitry that perform the task of the global inhibitor. As these elements have to be implemented with the equivalent of 10 gates (4 gates for the D-flip-flop have been subtracted to the total 14 gates per cell), it makes the digital approach at least as complex as the analog one.

All comparisons given above are not accurate due to the fact that performance of an algorithm strongly depends on its final application and the processing chain where it is embedded. LEGION computer simulations are best suited to investigate algorithm properties and possible improvements. G-LEGION scheme has proven its ability to segment high-resolution medical images if a powerful workstation is available and standard core microprocessors use peripheral circuits that makes it easy to embed in standard digital systems. Main advantages of our design compared to other approaches are its parallel nature that allows embedding light sensors with processing circuits and perform focal plane processing and its high speed while maintaining low power consumption and small die area. These are key benefits for portable systems.

IV.11 CONCLUDING REMARKS

In this chapter we have presented the results of the VLSI implementation of the segmentation network and analyzed its behavior under different biasing conditions.

We have demonstrated that oscillations are stable for each isolated oscillator and that oscillator periods through the network have a mean standard deviation on the order of 5% of the period. Interferences from other circuits and noise are small when compared to mismatch between oscillators.

Biasing currents control network behavior. They fix oscillation thresholds, frequencies, duty cycles and total power consumption, thus, network's ability to properly segment images. The most important biasing value in oscillation stability is discharging current because it is the lowest and usually operates transistors in sub-threshold region. Excitatory synapse current is also a key value because it accelerates oscillator shift from silent to active and vice-versa, reducing delays between cells. This reduction allows more objects to be set in each period.

Delays between oscillators have an important impact in network functionality. Some 'difficult' objects have been tested as spirals and they showed that indirectly coupled oscillators could be active at different intervals. Some kind of external process that groups oscillators –maybe based on inhibitor activity- seems necessary to solve that problem.

We also have given some boundaries on biasing currents and showed consequences of values beyond those boundaries.

When there is noise in the input image, the network showed that it works properly if appropriate biasing is used. However, as cells associated with noisy pixels do not oscillate, noise is not removed from the output, but simply ignored.

Finally, the network showed its speed and power consumption and these figures have been advantageously compared to other implementations. It has been demonstrated that it can segment objects very fast if excitatory connections are high enough and power consumption can be reduced to very low values.