

# Swimmy

v.24  
two-week tutorial  
for public release

Creative Commons Copyright



Dr. Franklin Krasne, 2008  
krasne@psych.ucla.edu

Editors: Natalie Schottler & Dr. William Grisham



## 1. General introduction

The goal of this laboratory exercise is to give you first-hand experience at analyzing a simple neural circuit that produces behavior. Specifically you will be observing the swimming movements and neural circuit of a virtual (i.e. a computer simulated) fish. You will be using a computer simulation to identify the neurons within the fish's nervous system that are responsible for this behavior and chart their excitatory and inhibitory neural interconnections within the circuit. The primary goal of this module is to determine the mechanism this circuit employs to produce the rapid left and right bends of the fish's tail that propel the fish through the water.

There are both practical and ethical reasons for using a realistic computer simulation in lieu of a live model. The ethical aspect is obvious; rather than subjecting a live fish to physiological experimentation, it is more humane to perform similar experiments using a computer simulation, which is feasible in this case. The practical reason for using a computer simulation is that it does not require a great deal of patience, precision, and time, which would be necessary in an animal experiment. Maintaining intracellular electrodes in small nerve cells while the animal executes locomotor activities, for example, is a skill that likely cannot be developed in the time allotted for this project. By doing computer-based experiments, skill development becomes unnecessary, yet pretty much the same experiments can be done.

Swimming and other rhythmic locomotor activities like flying, walking, or running are examples of universal behaviors that are essential for survival and therefore are important to study. Interest in the basis of rhythmic locomotor activities is related to important philosophical issues in the development of behavioral and neural science. As the scientific era of studying behavior began to emerge in the 19th century, there were two schools of thought about behavior. The *mechanistic approach* suggested that animals and their behaviors could be understood entirely in terms of physical and chemical laws. The contrasting *vitalistic approach* suggested that animals had unique attributes, some sort of special animate or "vital" force, that conflicted with the laws of physics and chemistry. Adherents of the vitalistic school were, in general, religiously oriented and tended to believe that patterns of behavior, at least in humans, were controlled by the hand of God. Adherents of the mechanistic school, while not generally atheists, felt that they could come to complete understanding of behavior without recourse to concepts such as vital spirit, soul, or will.

During the latter part of the nineteenth century, the principles of neural conduction and synaptic transmission were studied, and it became clear that behaviors such as reflexes (innate reactions to specific kinds of stimuli) could probably be fully understood in terms of neural circuitry and the newly discovered properties of synapses. However, much behavior does not occur in response to a stimulus, but is instead spontaneous. The basis for spontaneous behavior was not successfully resolved by nineteenth century physiologists, and because this type of behavior was most seen to be reflecting the action of a vital force, it proved a major challenge for the advocates of the mechanistic school.

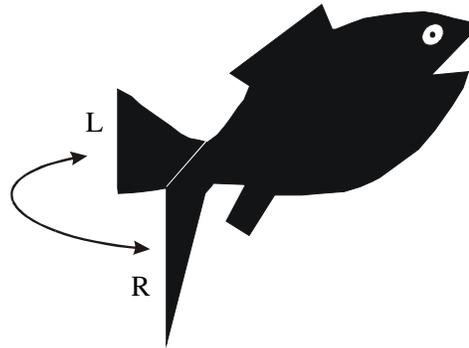
One of the most obvious and easily studied forms of spontaneous behavior is locomotion, in which rhythmic movements seem to be generated entirely by the animal itself. The study of the locomotion mechanisms thus became a surrogate for the study of spontaneous behavior in general. Many scientists felt that understanding locomotion would provide insight into the origins of spontaneous behavior, which could bear on the mechanism-vitalism debate.

There were three theories of locomotor production. The *chain reflex theory* proposed that rhythmic activities, such as repeated stepping, were the result of a chain reaction of reflexes in which an external stimulus initiated a behavior that in turn propagated a series of actions in response to the behavior that preceded it. The *endogenously rhythmic cell theory* proposed that certain neurons endogenously (intrinsically) generated rhythmic (bursting) patterns of activity, such as those involved in stepping or flapping wings. The *circuit theory* proposed that spontaneously rhythmic movements of locomotion were due neither to chain reflexes, nor to endogenously rhythmic cells, but were instead the result of interconnections of cells their individual and collective synaptic properties.

During the twentieth century, it was shown that nervous systems could continue to generate rhythmic patterns even after all possibility of sensory feedback was removed. Thus it became generally accepted that nervous systems were primarily responsible for producing normal, continuous locomotion without any sensory feedback at all. Chain reflexes, though not the basis for this behavior, often provided some modulatory control of locomotor patterns.

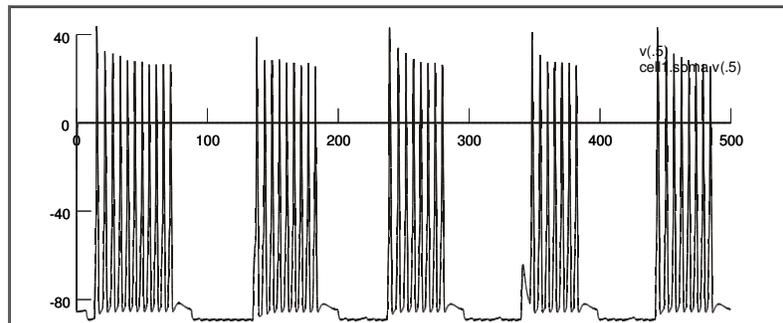
One of your tasks in this project is to determine which of three theories, detailed in **section 4.1**, can explain the rhythmic movements that constitute swimming in our computer fish.

Our computer fish is named *Swimmy*. *Swimmy* swims by flapping its tail rapidly from side to side, in a normal, continuous locomotive pattern that propels it through the water.

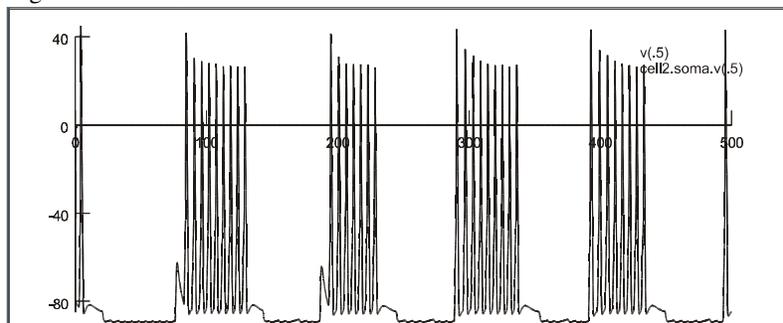


*Swimmy* has just one left-flexor motor neuron and one right-flexor motor neuron that innervates all of the muscle fibers of the left and right tail flexion muscles respectively. The recordings taken from both motor neurons are displayed below. There are alternating bursts of spikes, each lasting just under 100 msec, in the left and right motor neurons that are exactly out of phase with one another.

Left flexor motor neuron



Right flexor motor neuron



Your goal in this project is to determine how these motor neurons and other neurons in the swimming circuit interconnect and to explain how this particular pattern of behavior is achieved in the motor neurons. You can assume that production of rhythmic activity is due to one of the three mechanisms you will learn about when you reach **section 4.1** below.

## 2. Plan of Attack

There are two lab sessions for this project. *If you don't remember your neuron physiology basics, please review them to help you complete the better understand the project.*

- Week 1 Lab
  - Learn how to use NEURON, the program that runs Swimmy
  - Observe the activity of the motor neurons that allow Swimmy to swim
  - Observe basic neuronal activity, such as action potentials (spikes), EPSPs, IPSPs, etc., in action
  - Become familiar with neuronal and synaptic principles
- Week 1 Homework
  - Read section 4.1 and complete the Swimmy quiz, due at the start of lecture 2
  - Read section 4.2 and complete the given assignment, due at the start of lab
  - Read section 4.3 and complete the given assignment, due at the start of lab
- Week 2 Lab
  - Determine the interconnections between the swimming circuit neurons
  - Learn how to provide convincing evidence of direct neuronal connections
  - Determine which neurons generate the rhythmic pattern
  - Determine the type of oscillator these neurons employ
  - Learn how to provide convincing evidence for your conclusion about the origin and mechanism of the rhythm
- Week 2 Homework
  - Finish your analysis of the Swimmy circuit, as described in the lab manual
  - Complete the lab report in section 6, due at the start of the next lab session

## 3. Week 1 Lab

Swimmy is a virtual fish that exists in a simulation program called NEURON. The goal of this first lab is to guide you through the NEURON program and to review and solidify your knowledge of action potentials, synaptic potentials, etc.

### 3.1. Installing Swimmy

Swimmy is designed not to need installation of the sort that changes your computer registry. If you have access to any computer that runs the Windows operating system, you can either download and save Swimmy to any location on the computer or run it directly off a CD that has the program.

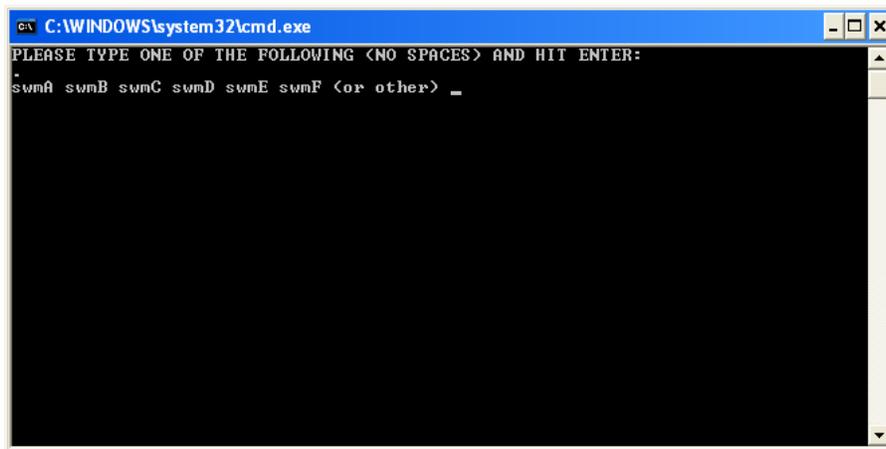
\*\*\* To properly download Swimmy from the website, see the sections titled [Preparing your computer for the SwimmyGUI download](#) and [Downloading SwimmyGUI to a Windows-equipped computer](#) in the **DOWNLOADING AND RUNNING SWIMMY GUIDE** at the end of this manual. \*\*\*

You can run the program from any location, including a CD. But to do this, first ensure that your computer has a **C:** drive (which the program occasionally calls to while it's running), then ensure that your computer has Microsoft .NET Framework Redistributable Package (a free downloadable component that will allow the animation to function properly).

### 3.2. Starting Swimmy

To access Swimmy you will need to locate the **SwimmyGUI** folder wherever you have downloaded it. Double-click the **SwimmyGUI** folder. Here you will find the **StartSWIMMY24** batch file that will initiate the Swimmy program. The **StartSWIMMY24** batch file contains everything you need to start your program.

To begin using the Swimmy program itself, go to the **SwimmyGUI** folder (if you are not already there) and double-click the **StartSWIMMY24** batch file.



A window will appear listing the six different student versions of Swimmy. Your instructor will inform you which of these programs you will be using, as listed below; circle the one you will be using for your own reference.

**swmA**  
**swmB**  
**swmC**  
**swmD**  
**swmE**  
**swmF**

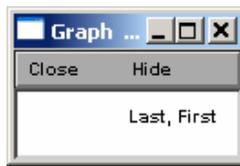
Type the program name in the window, then hit **Return** on the keyboard.

Once the program loads, six windows will be displayed. The **nrnalone** window will be referred to as the **Command Window**. You will probably not need to use the **Command Window**, but it must be open for NEURON to run. You may minimize the window so that it does not clutter your screen.

\*\*\* If these steps fail to initiate the Swimmy program, you may instead need to start Swimmy by entering a command in a command window, and then open the animation window separately. See the sections titled **Starting Swimmy using a command prompt** and **Initiating the Swimmy Animation window** in the **DOWNLOADING AND RUNNING SWIMMY GUIDE** at the end of this manual. \*\*\*



Position your mouse on the white line in the **NEURON** window and enter your first and last names. When you hit **Accept**, your name will appear in the blank graph. Reduce the dimensions of this window and drag it to the upper right corner of the screen for later use. This label will be required to appear on any printouts.



### 3.3. Displaying activity of Swimmy's motor neurons

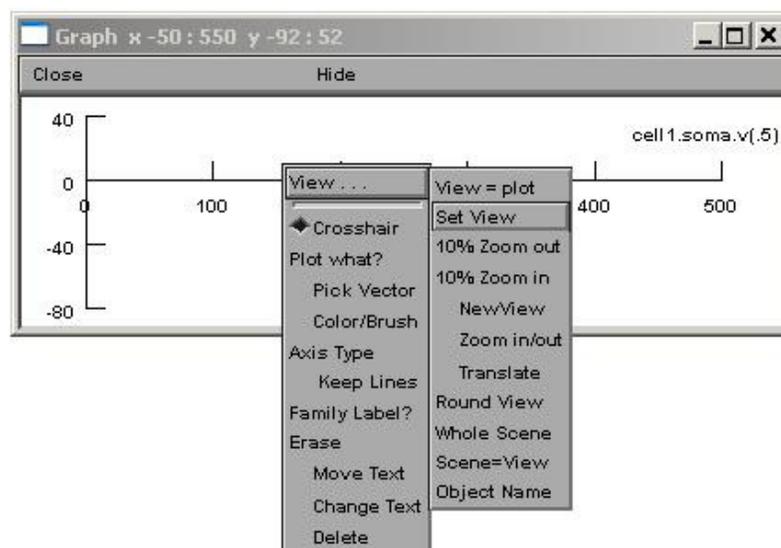
In order to visualize Swimmy's left and right motor neurons, named **cell 1** and **cell 2** in Swimmy, you must open up a graph that displays membrane potential as a function of time for each motor neuron. To do this, select **Make Record** from the **Swimmy Stuff** window. A list of various cells will appear below this button. Select **cell 1** to open the corresponding graph. Do this again to create a graph for **cell 2**. Position your graphs so that they don't overlap.

To display this activity of these cells on the graphs, click the **Init & Run** button. You will see alternating bursts of action potentials (spikes) in the two cells that resemble the activity diagrammed in **section 1**. In a real fish, this motor neuron firing pattern would cause alternating bends of the tail fin to the left and right. You should use your **Animate** window to better visualize this.

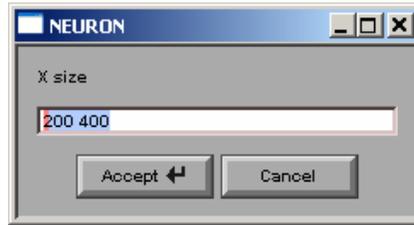
You may have expected spikes to always be of a fixed size, but you will notice that the first spike of each burst is larger than the rest. This is because the cells do not have time to entirely recover from refractory periods between spikes within a burst, and spike size is somewhat reduced in relatively refractory cells.

To view the neuronal activity more clearly, you may stretch your windows in any direction by dragging their edges. You may also zoom into your graphs by changing the scales of the x- and y-axes. Remember that the x-axis represents the time (in milliseconds, msec) that activity is occurring. Therefore when comparing neuronal activity, it is imperative that the x-axes of all neurons in question are identical in range. The y-axis represents membrane potential (in millivolts, mV) and changing the scale (the amount of zooming in or out) here is independent of other neurons. You may not necessarily want the y-axes of your graphs on the same scale; this depends on what you are trying to demonstrate.

To alter the scales of your axes, first right-click and hold the mouse at the center of the graph for **cell 1** and select **View**, then **Set View**.



Position your mouse pointer such that the tip lies within the space allotted for typing. Type in the following command to adjust the **X size** and hit **Accept**.

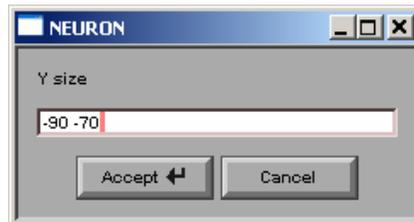


For now, do not adjust the **Y size** – simply hit **Accept** without changing the numbers.

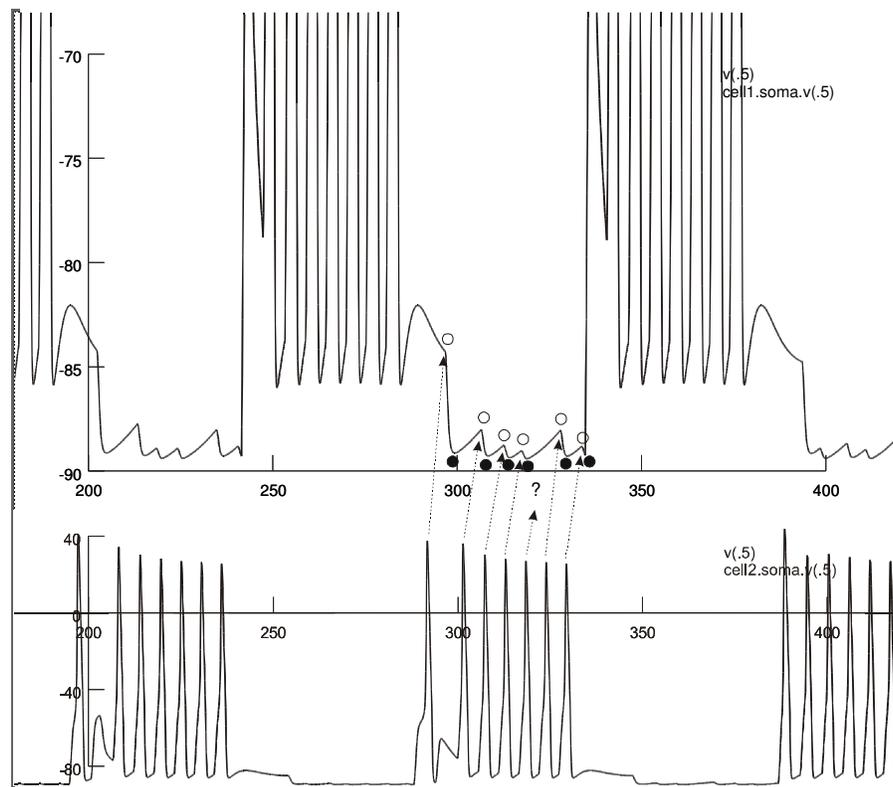


Repeat this zooming action for **cell 2**. Remember to only change the **X size** of **cell 2** for now.

Go back to **View...** → **Set View** on **cell 1**. This time adjust only the **Y size** by entering the following command.



This allows you to view the small between-burst events in **cell 1** more clearly.



Upon viewing this record for the first time, you might have considered an incorrect kind of interpretation. You may have thought that the low amplitude events in **cell 1** (between about 300-340 msec) were a series of *depolarizations* (presumably EPSPs) whose peaks are marked by the open circles. If you recall the IPSPs that you have seen in textbooks and classes, they always have relatively rapid onsets and slow decays, and never the reverse (the same is true for EPSPs). Therefore, the activity in **cell 1** cannot be characteristic of EPSPs, but are due to IPSPs instead.

We can correctly interpret these graphs as follows: The spikes seen in **cell 2** are followed, at a short delay, by hyperpolarizations, presumably IPSPs, which start as indicated by the dashed arrows and which *peak* at the solid dots. Although the **cell 2** spikes *usually* cause an IPSP, they do not always do so, as shown by the missing IPSP indicated by the question mark. The fact that IPSPs in **cell 1** usually follow spikes in **cell 2** but do not *necessarily* do so would lead us to conclude that the IPSPs are caused by an interneuron that is excited by **cell 2** and that sometimes fails to fire in response to a spike in **cell 2**. According to Dale's Law, the observation that a given neuron releases the same neurotransmitter at all open terminals also argues for the existence of an inhibitory interneuron. We know that **cell 1** and **cell 2** are motor neurons and therefore must release acetylcholine, which is not usually inhibitory.

You might also question, when observing the above figure, why the first IPSP in the series is so much larger than the others. Two explanations are available:

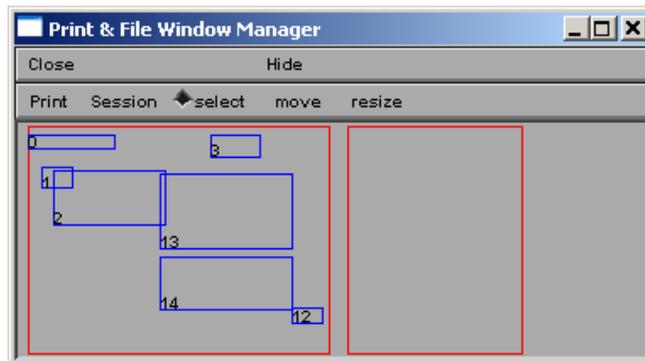
- (1) IPSPs get smaller because the inhibitory synapses become depressed by ongoing activity (i.e. they act as though they tired out or fatigue).
- (2) Inhibitory input to the cell causes the membrane voltage to move toward about -90 mV, and since the membrane potential is almost all the way there when the second inhibitory input occurs, relatively little additional downward shift occurs due to that input. This topic will resurface later in this manual.

For now it is sufficient to conclude that Swimmy is able to swim because his motor neurons alternate between producing bursts of spikes and being inhibited. Moreover, since each IPSP follows at a fixed time after a spike in the contralateral motor neuron, it looks as though the IPSPs are probably produced by the spikes in the contralateral motor neuron, but via an interneuron.

### 3.4. Printing out records

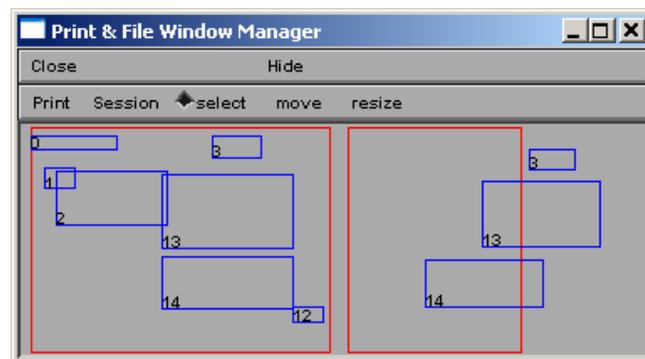
To ensure that the windows you wish to print will fit on the printed sheet of paper, make sure that these windows are not more than about half the screen wide. If they are not, resize them by dragging their edges with the mouse pointer.

Click **Window** in the **NEURON Main Menu** and select **Print & File Window Manager**. This will bring up a window with two red rectangles. The left rectangle is a map of the windows, in blue, on your desktop. The right rectangle is a map of the windows that will appear on your printout. **NEVER CLICK THE RESIZE BUTTON IN THIS WINDOW. IT CAN DESTROY THE DIMENSIONS OF THE WINDOWS ON YOUR SCREEN AND CAUSE SOME OF THEM TO BE HIDDEN OFF THE SCREEN.**

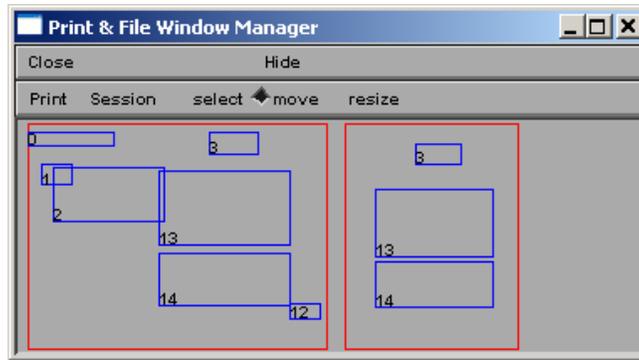


Follow your instructor *very carefully* to complete the following details.

Make sure the black diamond appears next to **select**, otherwise click this button. Select the windows you wish to print by single-clicking the corresponding blue windows in the left red rectangle. For your practice, select the windows representing **cell 1** and **cell 2**. Be sure to include the window that has your name on it. They will be duplicated on the right side of the **Print & File Window Manager**, but may not necessarily appear within the lines of the right red rectangle. **THIS IS OKAY!!!**



If your blue windows are not contained within the right red rectangle, click the **move** button so that the black diamond now appears here. Use your mouse to drag your selected windows into the right red rectangle.



If your blue windows are too long to fit in the right red rectangle, **DO NOT RESIZE THE RECTANGLES IN THE PRINT WINDOW BOX!!!** Instead, simply resize the original windows on the desktop by dragging the edges with your mouse.

When you have finished rearranging your selected windows, click **Print** and then click **To Printer** in the drop menu. Accept any additional windows that may pop up.

If you are having trouble printing using Swimmy's print option, you can instead press the **PrintScr** button on your keyboard to capture a screenshot of your computer desktop, paste the image into a word processing program (like Microsoft Word or OpenOffice SWriter), and print out the document.

For cleaner screen images, you can capture each window individually by selecting the window and pressing both **Alt** and **PrintScr** simultaneously on your keyboard and pasting each into a word processing program for printing.

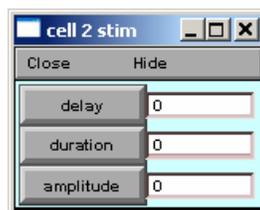
Once printing is successfully completed, adjust the **Y size** of your **cell 1** record so that it displays membrane potential from  $-80$  to  $40$  mV. Leave both the **cell 1** and **cell 2** windows open to use for the following section.

### 3.5. Stimulating by passing current through an intracellular electrode

An experimental technique that is standard fare for real neurophysiologists and that will be essential for this exercise is induced stimulation of the neuron. Using this technique will allow you to do either of the following:

- (1) depolarize neurons in order to force them to fire
- (2) hyperpolarize neurons in order to prevent them from firing

You can arrange to pass current into a neuron by selecting **Make Stim** from the **Swimmy Stuff** window. Select the cell you wish to stimulate from the drop menu. For this example, select **cell 2**. A **cell stim** window will appear for the corresponding cell.

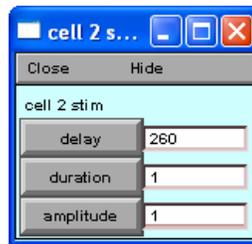


The **cell stim** window has three features:

- (1) The **delay** represents the time (in msec) that you wish to begin your stimulation. Remember that time is displayed on the x-axis. Therefore, your delay is the time on the x-axis where your stimulation will begin to affect the cell.

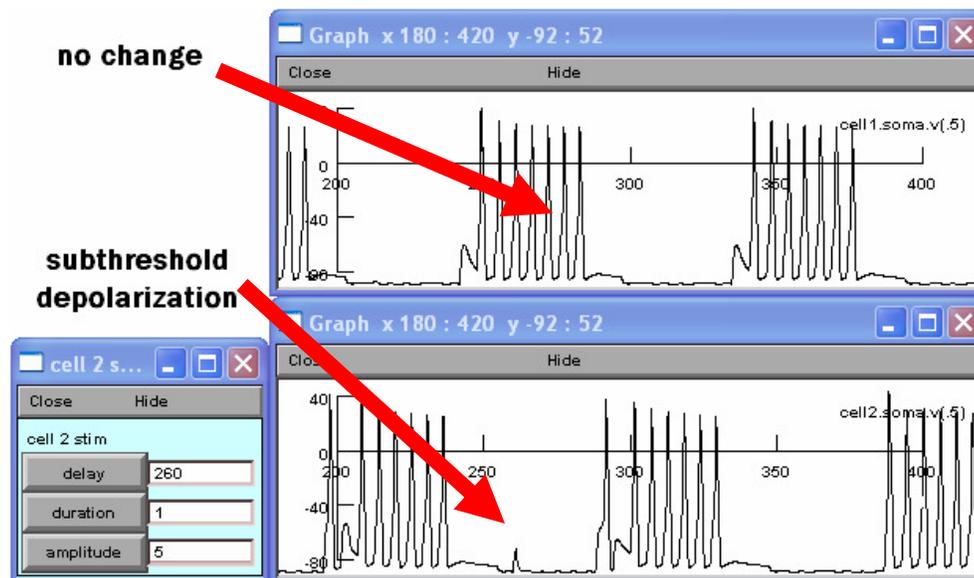
- (2) The **duration** is the length of time (in msec) that you wish to maintain your stimulation. Therefore, your cell will be stimulated starting at the delay (in msec) you chose above, and will last for the duration (in msec) you enter here.
- (3) The **amplitude** of the current represents the strength (in nano-amps) of the stimulation. Positive amplitudes will depolarize the cell. Negative amplitudes will hyperpolarize it. The larger the absolute value of the amplitude is, the stronger the current is. Try not to exceed +/- 15 nano-amps unless it is absolutely necessary to use a current that powerful.

To generate a 1 msec depolarizing current pulse during one of the intervals between spike bursts in **cell 2**, find a region on the **cell 2** graph where such an interval occurs. Two areas exist: the first is around the 260 msec region, and the second is around the 360 msec region. We will arbitrarily pick the 260 msec region to view our stimulation. In your **cell 2 stim** window, enter the following information.

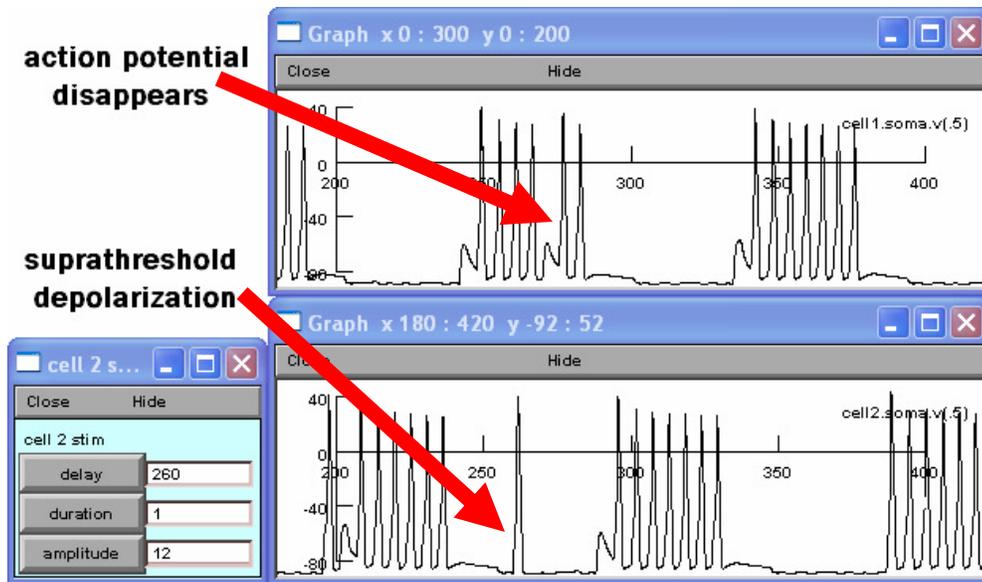


This means that our stimulation will initiate at 260 msec on the x-axis and will last for only 1 msec. When you begin with an amplitude of only 1 nano-amp you will observe a small depolarization around 260 msec on your **cell 2** record. This shows that your pulse was not strong enough (it did not have a large enough amplitude) to depolarize the cell to threshold (to generate an action potential).

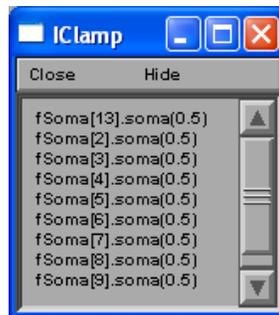
Gradually increase the amplitude in about 1 or 2 nano-amp increments until you find (approximately) the minimum amplitude needed to allow this cell to fire at the given delay. Hit **Init & Run** each time you change the amplitude value. Watch how as you increase the amplitude, the amount of depolarization also increases.



Once you find an amplitude that allows **cell 2** to fire (a supra-threshold depolarization), you may notice that a spike that occurred just after 260 msec on the x-axis of **cell 1** *no longer exists*. This evidence strengthens the interpretation, as suggested above, that spikes in one motor neuron cause IPSPs in the opposing motor neuron, probably via an inhibitory interneuron.



You can establish as many stimulus channels as you like in a cell, and each will deliver a current pulse (with the characteristics specified) when you hit **Init & Run**. When you are done with a stimulus you can remove it by first setting the **amplitude** value to zero, then simply closing or minimizing the **cell stim** window. **IT IS IMPERATIVE** that the **amplitude** is set to zero before closing it. If you fail to do this, the cell will *remain stimulated* at the given point in time and you will not be able to control it. If you forget and end up having a problem, select **Tools** in the **NEURON Main Menu**, then **Point Processes**, then **Viewers**, then **Point Processes**, then **IClamp**. An **IClamp** window will appear containing a list of **cell stim** windows that have been used in the program thus far.



Double-clicking any of these items in the list will re-open the corresponding **cell stim** window. You should NEVER open the very **FIRST** item in this list. In some versions of Swimmy the first item is labeled **fSoma[13].soma(0.5)**, and in other versions of Swimmy the first item is labeled **fSoma[14].soma(0.5)**.



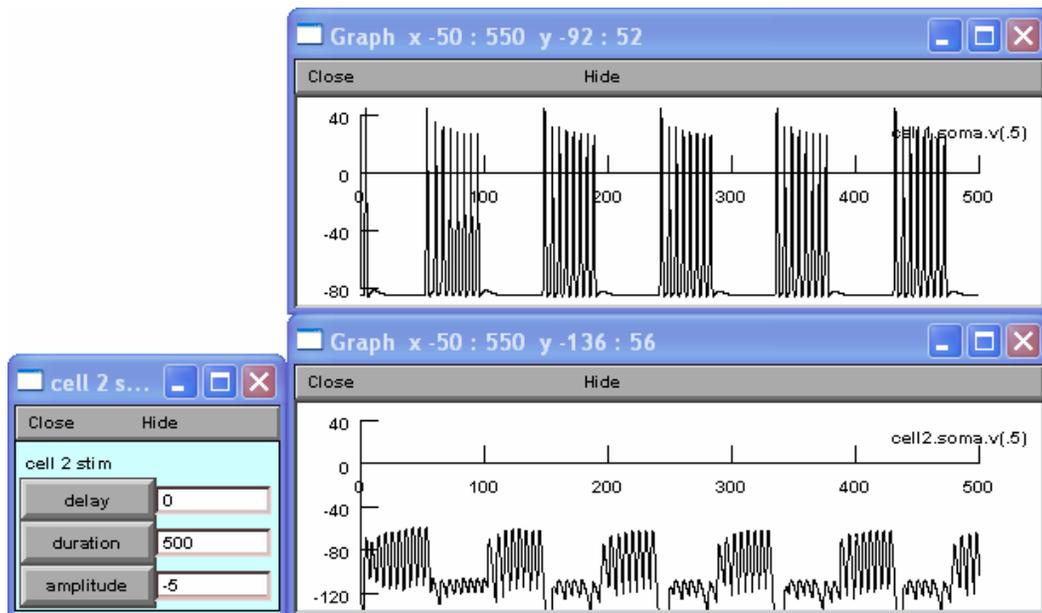
This first item in the **IClamp** window corresponds to a pre-programmed stimulation that helps the Swimmy circuit to function properly. It should not be opened and subsequently adjusted. You are welcome, however, to open any of the

remaining list items to access a **cell stim** window that you have used before. Note that the numbers in the list DO NOT always correspond to the cell number that was stimulated. This means that you may have to re-open most of the **cell stim** windows in the list to find out which cells still have lingering stimulation. Make sure to zero any amplitudes that have other values in them, hit **Init & Run** to ensure that you have disabled the stimulation, and close the **cell stim** windows. Ask your instructor for help if necessary.

You will still need to use your **cell 2 stim** window, so do *not* zero out and close this window yet. You will be instructed to do so later on.

You can now use your ability to stimulate cells to find out something about Swimmy that will be useful later on. Swimmy bends alternately left and right as the left and right motor neurons burst in alternation. If one of the motor neurons were silenced by hyperpolarizing it, what would happen to Swimmy's ability to swim? After bending its tail to one side, would the tail relax back to the midline position as a result of some sort of elasticity in the body, ligaments, or muscles, or does the tail need the contraction of the muscles on the other side to pull it back? To answer this question silence one of the motor neurons (**cell 1** and **cell 2**) and animate Swimmy to see whether or not its tail relaxes back to the midline in between bursts of the other motor neuron or if it instead deflects to one side.

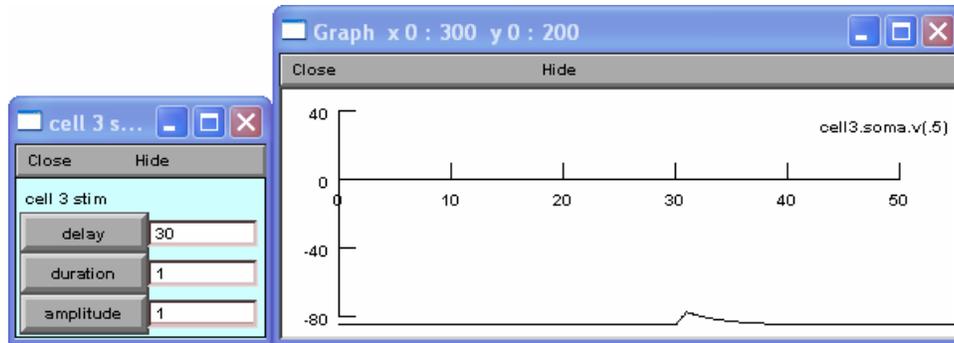
To carry out this experiment, first adjust the **X size** of your **cell 1** and **cell 2** records so that they display 0 to 500 msec of activity. Then hyperpolarize **cell 2** for the duration of the record (0 to 500 msec). This means that you must change the values in your **cell 2 stim** window, so that the delay is 0 msec, the duration is 500 msec (which will cause the stimulus to last until 500 msec), and the amplitude must be a negative value to hyperpolarize the cell. After you hit **Init & Run**, you may need to adjust the **Y size** on your **cell 2** record in order to see the activity that occurs in this cell. Although the hyperpolarizing stimulus you gave the cell will prevent its firing, it is still receiving input from other cells and therefore still exhibits sub-threshold activity. Your windows should look like those in the figure below.



Once you have established the stimulation, click **Animate** in the animation window and observe the resulting tail movements. The animation window displays 500 msec of activity, so look for Swimmy to alter its tail movements throughout the entire animation. Closely observe Swimmy's behavior in this window and take detailed notes regarding its pattern of movement. Be prepared to contrast this behavior with another type of activity listed in **question 9** of your lab report.

At this time, you may zero your amplitude in the **cell 2 stim** window, hit **Init & Run** to ensure that you have disabled the stimulation, and close the **cell 2 stim** window. You may also close your graphs for **cell 1** and **cell 2**, since the following section does not require them.

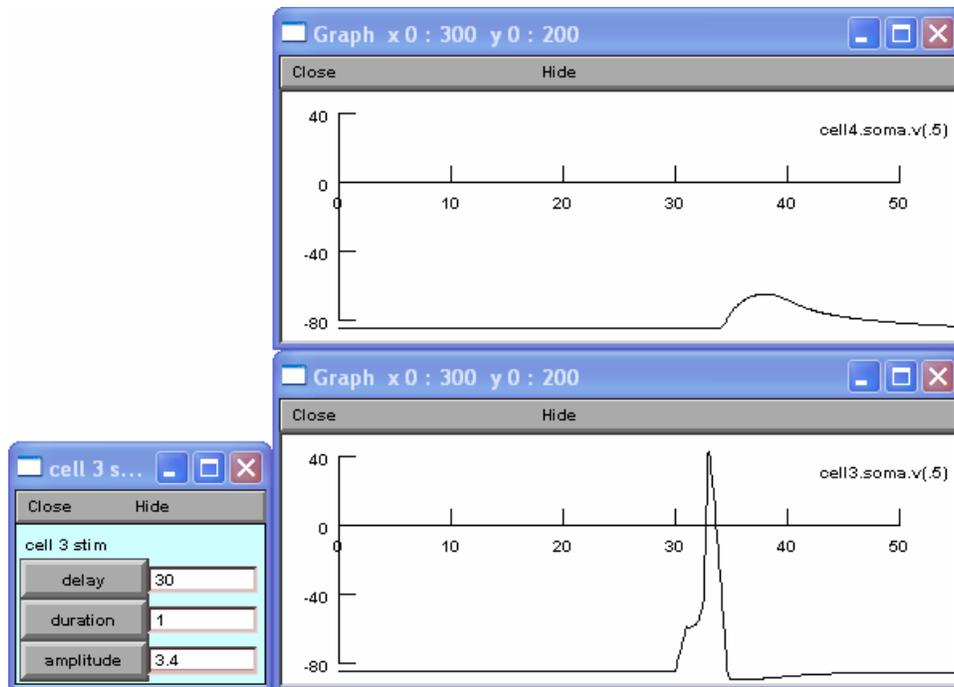




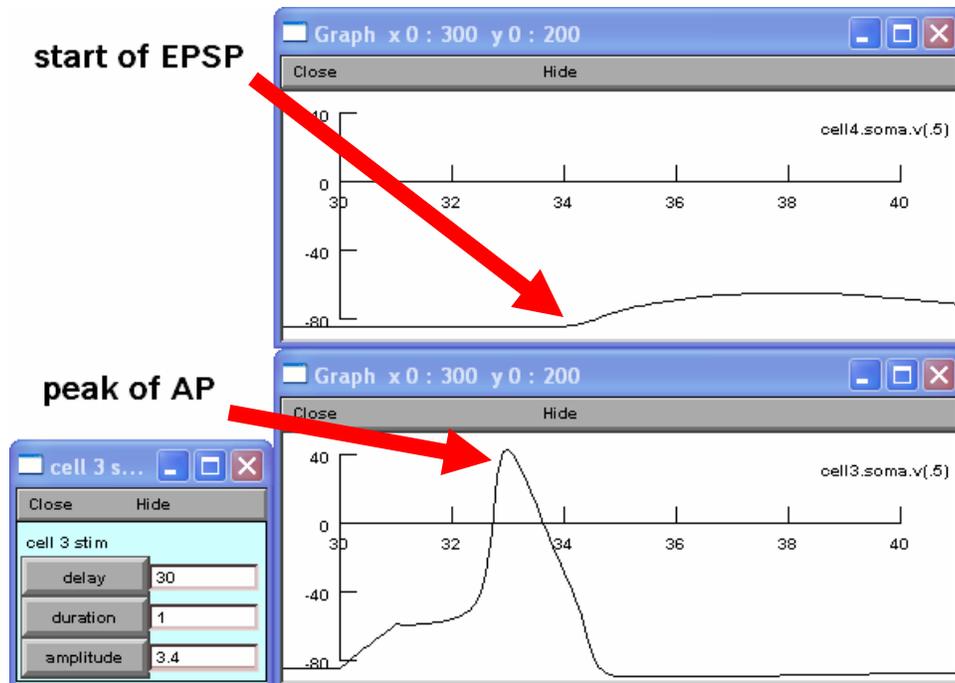
You should find that as you gradually increase the amplitude of a depolarizing current stimulus, the depolarization in **cell 3** will increase the membrane potential until the point that it hits threshold and fires (generates an action potential). Determine the threshold current to within 0.1 nA (continue increasing the amplitude by 0.1 until you see the amplitude required to make **cell 3** spike), and see for yourself whether the spikes are really all-or-none, as they are supposed to be.

### 3.6.1.2. Excitatory postsynaptic potentials

Change the time axis (x-axis) for **cell 4** so that it is the same as that of **cell 3**. Use the **cell 3 stim** window to stimulate **cell 3** just above threshold. Note that an EPSP appears in **cell 4**. This EPSP is too small to cause cell 4 to fire (it is sub-threshold). Note also that the EPSP *begins* about 1 msec after the spike in **cell 3** reaches its *peak*.



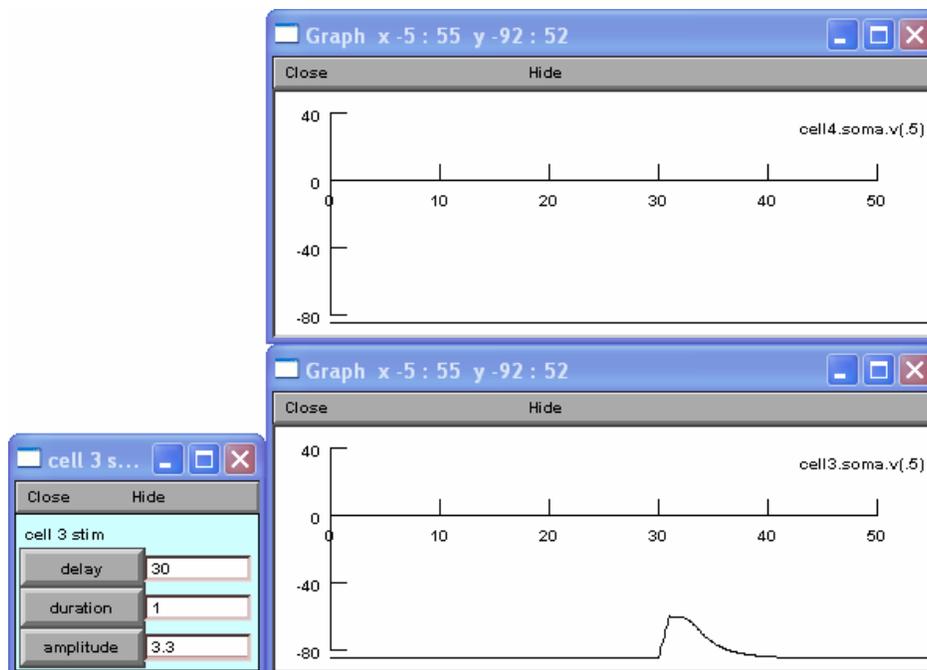
To better observe the 1 msec delay, you may want to *temporarily* zoom into your time axes so that they both run from just 30 to 40 msec.



This delay, sometimes called the “central delay,” is the same for all Swimmy neurons; you can take advantage of this fact later to decide whether one cell affects another either directly or indirectly through one or more interneurons. The central delay is composed of the intrinsic synaptic delay plus a little bit of time for conduction from the soma (where you are recording) to the synaptic terminals of the cell (we have made conduction times in Swimmy negligibly short).

**Zoom your graphs back out to display 0 to 50 msec on the x-axis, and continue with the following procedures.**

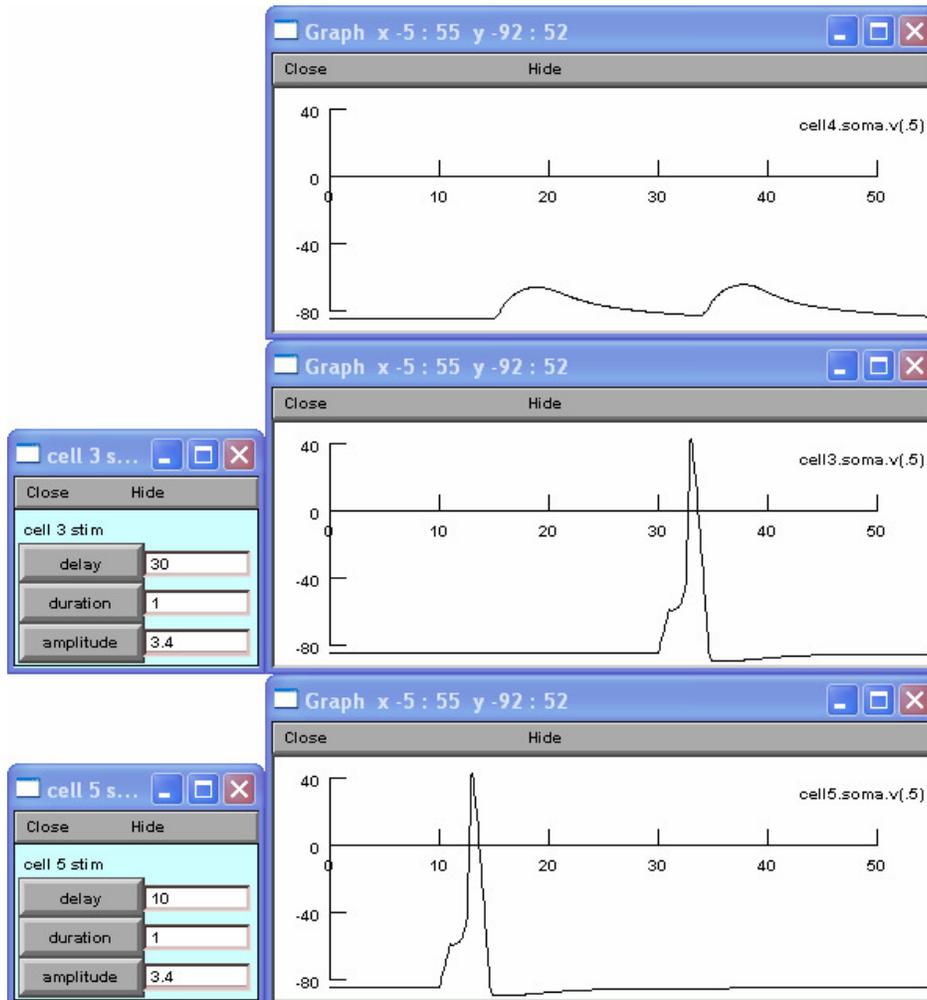
Change the amplitude in the **cell 3 stim** window so that **cell 3** is below threshold (it does not fire). Note that when a presynaptic neuron (here, **cell 3**) does not fire, it is unable to send any signals to the postsynaptic neuron (here, **cell 4**).



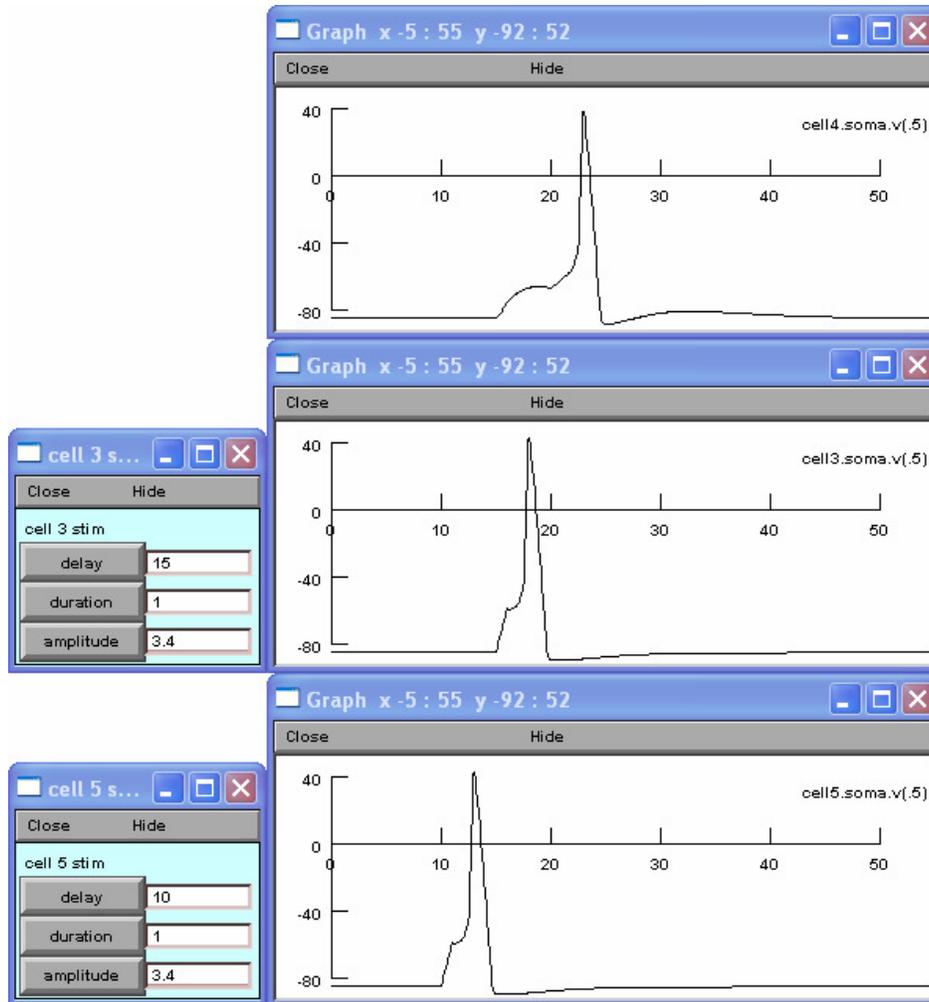
You may leave your **cell 3 stim** window open to continue with the following procedures.

### 3.6.1.3. Summation

As shown in the diagram for **circuit (a)** on page 16, **cell 5** excites **cell 4**. Open a **cell 5 stim** window and give the cell a 1 msec long supra-threshold (above threshold) stimulus at 10 msec. Adjust the amplitude in the **cell 3 stim** window so that it also is supra-threshold. Your windows should look like those shown below.



You can see that when **cell 5** fires, it produces an EPSP in **cell 4** that is equal in size to the other EPSP in **cell 4** produced when **cell 3** fired. If you gradually change the delay in the **cell 3 stim** window so that the stimulation begins earlier, you will see the effects of summation. To do this, decrease the delay in increments of 5 msec, hitting **Init & Run** each time, until the two EPSPs in **cell 4** combine enough to cause the cell to fire.

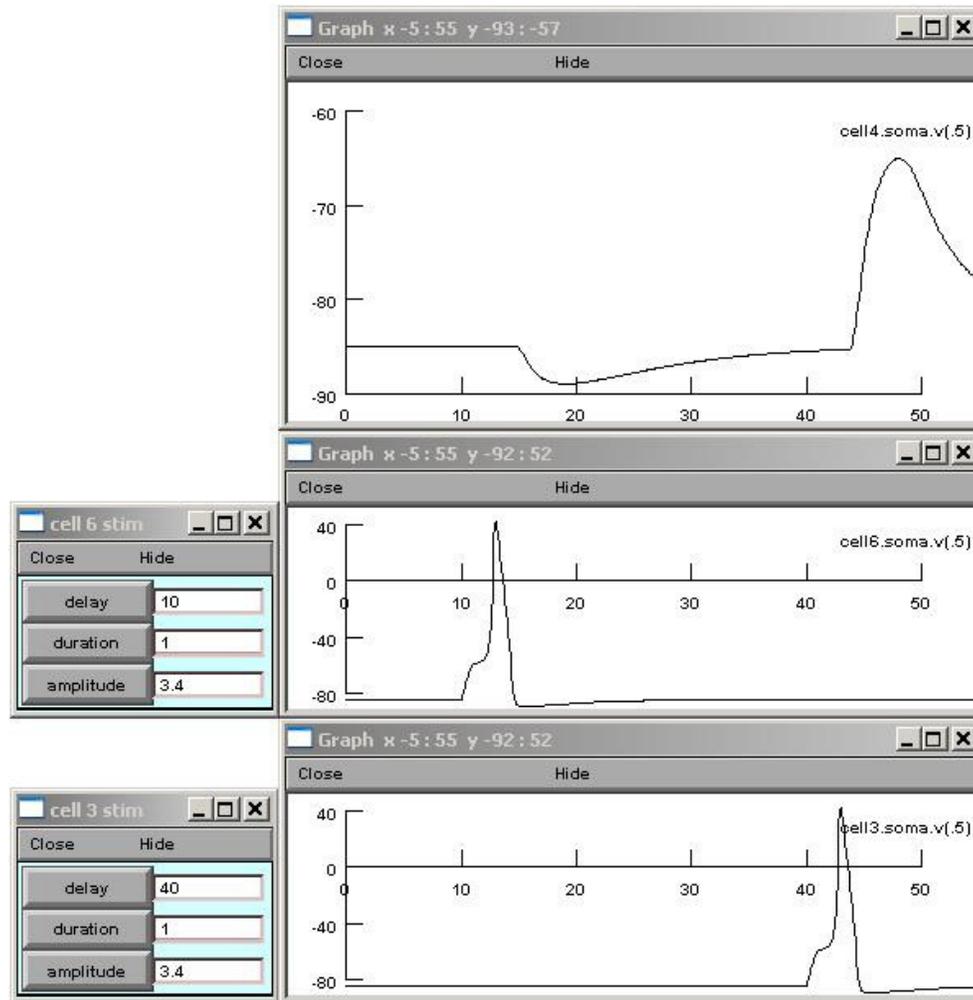


This is an example of spatial summation *and* an example of temporal summation because the two EPSPs can sum enough to fire **cell 4** even though they occur at *different* times.

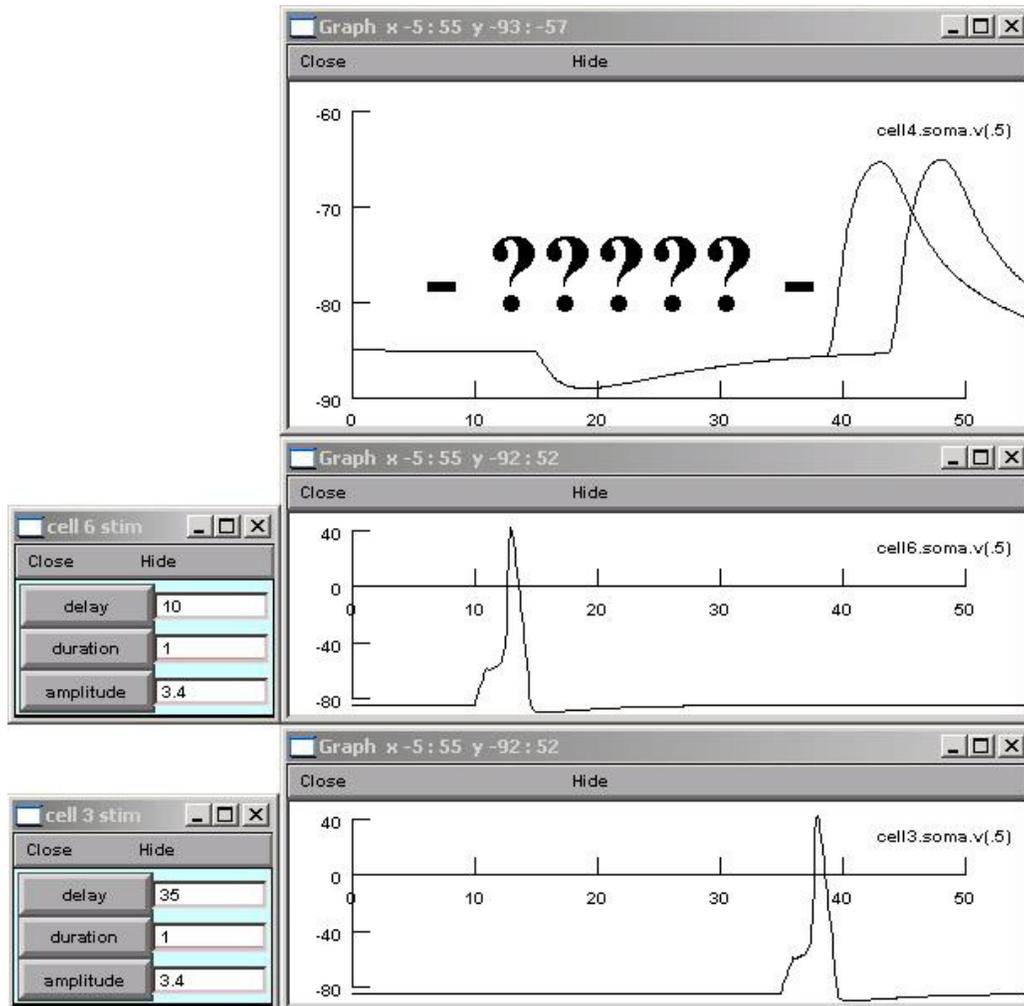
When you have completed this exercise, zero out your amplitude in the **cell 5 stim** window, hit **Init & Run** to make sure the stimulation has been discontinued, and close the **cell 5 stim** window for now. You will still need your **cell 3 stim** window open for the following sections.

#### 3.6.1.4. Inhibition

As shown in the diagram for **circuit (a)** on page 16, **cell 6** inhibits **cell 4**. To examine this inhibitory effect, open a **cell 6 stim** window and give the cell a 1 msec supra-threshold depolarizing pulse at 10 msec. Adjust the delay on your **cell 3 stim** window so that its stimulation begins at 40 msec. You will see two different kinds of activity in the record of **cell 4**. The first is an IPSP that begins about 1 msec after **cell 6** fires. The second is an EPSP that begins about 1 msec after **cell 3** fires. You may need to zoom into your record of **cell 4** (namely, the y-axis) to better visualize this.



In this instance, it may be helpful to keep old traces on the graphs when you produce new ones. To do this, right-click the center of the record for **cell 4** and select **Keep Lines**. Gradually change the delay in the **cell 3 stim** window so that the stimulation begins earlier. To do this, decrease the delay in increments of 5 msec, hitting **Init & Run** each time, until **cell 3** is being stimulated at the same time that **cell 6** is (at 10 msec). You should observe the activity displayed in the following figures.



As the EPSP in **cell 4** occurs progressively earlier, the maximum absolute voltage it achieves becomes lower and lower. Thus, due to its interaction with the IPSP, the EPSP is getting smaller and smaller in amplitude (height, or distance of the deflection from baseline to peak). Nonetheless, there is another thing to notice. Measure the amplitude of the EPSP and the IPSP when they are *not* summated, as well as when they *are* summated (when the two PSPs occur at the same time). Left-click and hold the mouse cursor on the trace and observe the y-axis value listed at the top bar of the graph. Use this value to calculate the amplitude of an EPSP or IPSP by comparing the point of maximum deflection to the y-axis value of the baseline (when there is no EPSP or IPSP). Be prepared to discuss what you are seeing here in your lab report for **question 1**.

To proceed with the following section, zero the amplitude on the **cell 3 stim** window, hit **Init & Run** to ensure the stimulus has been removed, *but leave the window open*. You will be using your **cell 3 stim** window in the latter half of the following section. Right-click the center of the **cell 4** record and select **Erase** to remove the traces from this experiment. You will need **Keep Lines** activated for **cell 4**, as well as both the **cell 3 stim** and **cell 6 stim** windows open for the next section.

### 3.6.1.5. Effects of altering membrane potential on IPSPs

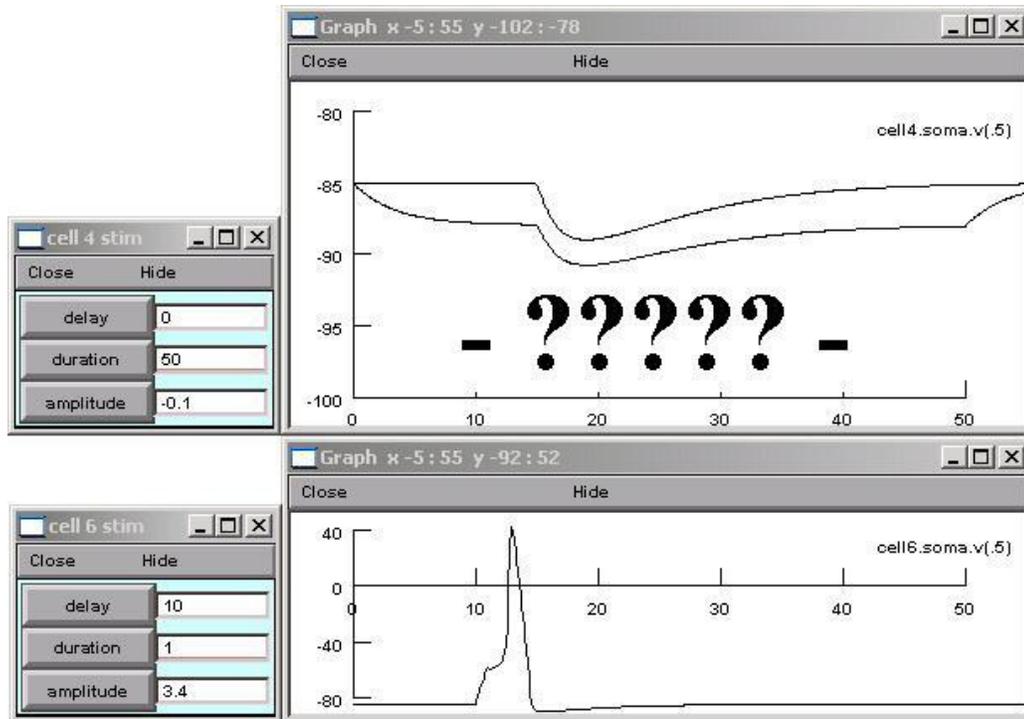
In certain cases, you will want to stop a cell from firing (by hyperpolarizing it) so that you can see what effect this has on the operation of the swimming circuit or so that you can see EPSPs and IPSPs arriving at a cell without the confusion produced by spikes. You can do this by passing into a current into the cell, of negative amplitude, that hyperpolarizes it. This prevents the cell from reaching its critical firing level. We will practice this technique shortly.

Experiments in which you hyperpolarize cells can be very useful. However, hyperpolarizing and depolarizing *postsynaptic* cells can produce some odd effects on both EPSPs and IPSPs that can confuse you if you are not prepared for them. These effects result when synaptic activation of the hyperpolarized postsynaptic neuron causes the membrane potential to move toward the reversal potential defined by the mix of channels opened by the neurotransmitter. The reversal potential is that membrane potential at which there is *no net movement of ions* across the membrane and therefore no current. Because of this, the size and direction of synaptic potentials changes as a function of the baseline membrane potential.

The stronger the hyperpolarization (the lower the membrane potential) of a postsynaptic cell that is also receiving *excitatory* input, the larger the resulting EPSP amplitude (height) measured from the baseline potential. As the membrane potential gets farther below the reversal potential, the driving force for sodium influx becomes greater, thus giving a taller EPSP.

When progressively hyperpolarizing a cell that is receiving *inhibitory* input, the resulting IPSP will decrease in amplitude, flatline at the reversal potential, and invert once the membrane potential is below the reversal potential. Relatively strong hyperpolarizations render the cell's interior extremely negatively charged. If the inside of the cell becomes more negative than the reversal potential associated with ions to which the cell becomes permeable during inhibitory action, negatively charged ions will be driven out of the cell or positively charged ions will be driven in. Accordingly, if the cell is hyperpolarized below the reversal potential for inhibition, it will become *positive* during the action of the inhibitory neurotransmitter.

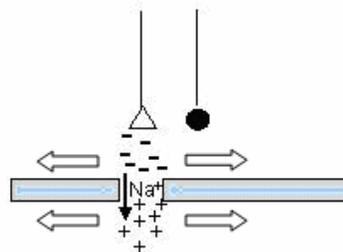
To see an example of the effect just described, arrange your stimulation so that your inhibitory neuron, **cell 6**, fires all by itself at about 20 msec for a duration of 1 msec. Alter your **cell 4** y-axis so that it runs from -100 to -80 mV (this will make it easier to see your IPSP well) and make sure the **Keep Lines** function is checked for **cell 4**. Use the **Make Stim** button to create a **cell 4 stim** window then examine the effects of hyperpolarizing this cell (**cell 4**) at a delay of 0 msec for a duration of 50 msec at an amplitude of -0.1 nA. Continue hyperpolarizing the cell in 0.1 nA increments, until you have reached a current strength of -0.5 nA. You will see that as you strengthen the hyperpolarization of **cell 4**, the IPSP decreases in size and then flips upside down after the membrane potential has met the reversal potential. The result is an IPSP that looks instead like an EPSP. Remember that even though it *looks* like an EPSP, you must *call* it an IPSP because the activity is *still* a result of opening of postsynaptic membrane channels that were designed to cause inhibition, not excitation. Indeed, there are *some* animal synapses (though not any of Swimmy's) in which the inside and outside concentrations of chloride (the ion to which the membrane usually opens when inhibitory transmitters are released) are such that the IPSP is in fact slightly *depolarizing* rather than hyperpolarizing, but since the IPSP is well below the critical firing level, it never causes firing.



### 3.6.1.6. Effects of shunting on membrane potential

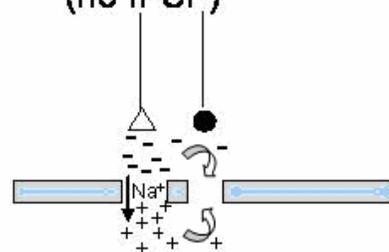
Now that you are familiar with reversal potentials, we can further explore the true nature of inhibition. An important thing to understand about inhibition is that one cannot simply subtract the hyperpolarizing IPSP from the EPSP. As you saw in **section 3.6.1.4**, an EPSP actually *shrinks* in size when it occurs during inhibitory action. To see why this is so, suppose that we have hyperpolarized **cell 4** so that it is sitting *exactly* at the reversal potential for inhibition. During inhibitory transmitter action, channels in the **cell 4** membrane would open to permit ion influx, but there would be *no net flow* of positive or negative ions, as indicated by the absence of a deflection in membrane potential characteristic of a PSP. The open membrane channels *do* have an effect however. They increase the electrical conductance of the membrane (i.e. they *lower* its electrical *resistance*, since resistance and conductance are reciprocals of each other) at the inhibitory synapse, and this causes a sort of short-circuiting of the membrane. The following picture should help you understand what happens.

(A) Without inhib



Charge moves through membrane and spreads laterally.

(B) With inhib  
(no IPSP)

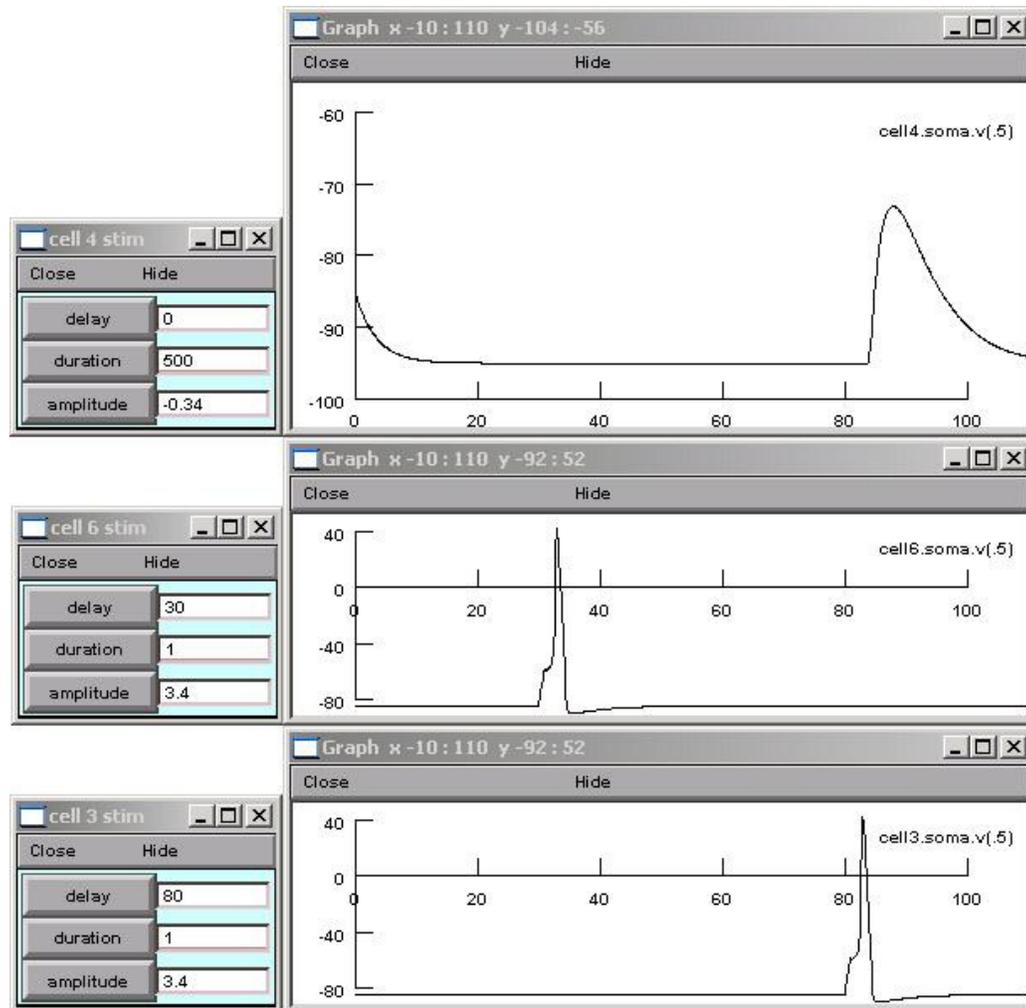


Some charge moves back across membrane through "hole" (Cl<sup>-</sup> channel) instead of spreading to rest of cell.

To show that this really works, right-click the center of the **cell 4** record and select **Erase** to remove the traces from the previous experiment. Right-click the window again and *deselect* the **Keep Lines** function. Adjust the **X size** of the **cell 4**, **cell 3**, and **cell 6** windows to display activity from 0 to 100 msec. Adjust the **Y size** of the **cell 4** window to display membrane potential from -100 to -60 mV. Then set the duration on your **cell 4 stim** window to 500 msec.

Stimulate **cell 6** so that it fires at a 30 msec delay for a duration of 1 msec. Stimulate **cell 3** so that it fires at 80 msec delay for a duration of 1 msec.

Adjust the amplitude (by making it negative) to find a hyperpolarization level that causes your IPSP to completely vanish. This amplitude should fall between 0 and -0.5 nano-amps, as seen in the figure below. Remember that the inhibitory transmitter action is *still there* at 30 msec.



Right-click the center of your **cell 4** record and again select the **Keep Lines** function. Now gradually move your stimulation of the excitatory input earlier, as you did in the previous section. What happens when the EPSP occurs at the same point in time as the hidden inhibition? Measure it when induced at 80 msec and at 30 msec. Be prepared to address this issue in your lab report for **question 1**.

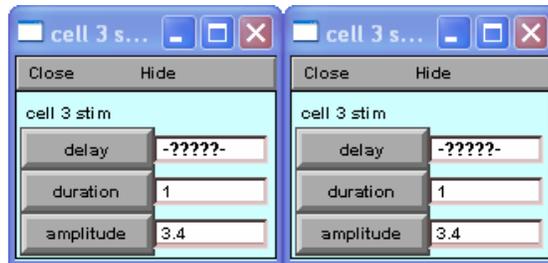
You may now unclick the **Keep Lines** function and use **Erase** to clear the graphs. Zero the amplitudes on the **cell 4 stim**, **cell 6 stim**, and **cell 3 stim** windows, hit **Init & Run** to confirm this, and close these windows.

### 3.6.17. Short-term plasticity

Synaptic strengths undergo long-term change due to processes like LTP and LTD. None of Swimmy's synapses are capable of either of these. Synapses can also often undergo short-term changes, lasting in the 10-100 msec range or longer, that serve a variety of information processing functions. This short-term plasticity may be found in Swimmy. Because such short-term plasticity plays a role in some theories about the generation of motor patterns like walking and swimming, we need to discuss it briefly.

Sometimes the effectiveness of a synapse increases. In other words, the effect of activating a synapse can strengthen subsequent activations that shortly follow it, which is called *facilitation*. Other times, the effectiveness of a synapse decreases, which is called *depression*.

Both the excitatory synapses between **cell 3** and **cell 4** and between **cell 5** and **cell 4** demonstrate short-term plasticity. Make sure that you have a record open for **cell 3**, **cell 4**, and **cell 5** but only manipulate ONE synapse at a time (**cell 3 & cell 4** OR **cell 5 & cell 4**). One synapse exhibits facilitation and the other exhibits depression. It is up to you to determine the plasticity of these two synapses. To do this you will need to establish two stimulus channels for **cell 3** when observing the synapse between **cell 3** and **cell 4**. You will also need two stimulus channels for **cell 5** when observing the synapse between **cell 5** and **cell 4**. For each presynaptic cell, arrange one of the **cell stim** windows to produce a current pulse that fires the cell once (for about 1 msec), then arrange the other to produce a second current that fires the cell at a short period of time afterward. You need to choose an appropriate time interval to stimulate the presynaptic cell so that the synapse will demonstrate facilitation or depression WITHOUT demonstrating summation.



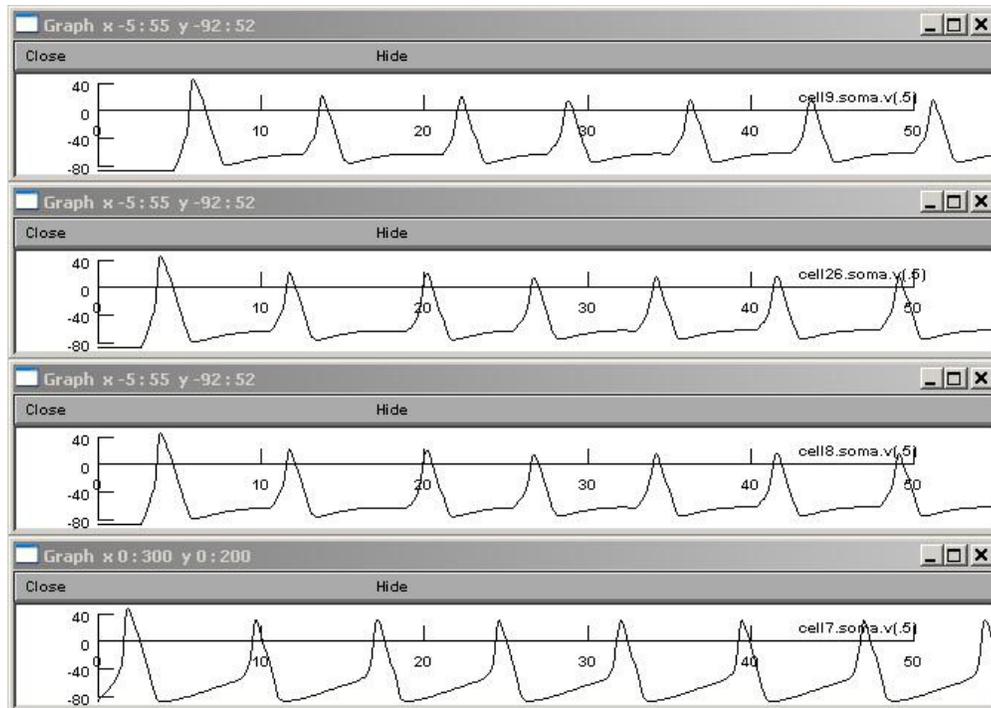
After you examine the effect these spikes have on the postsynaptic cell, **cell 4**, print your evidence of plasticity to use in your lab report for **question 2**. Be sure to include your **station name** and **cell stim** windows along with your records. When you have done this, you may zero the amplitudes on all **cell 3 stim** and **cell 5 stim** windows, hit **Init & Run**, and close the **cell stim** windows. You may also close the records of all the cells in **circuit (a)**, since they have no involvement in the remainder of this project.

### 3.6.2. Circuit (b)

Begin by opening a record for all the cells in **circuit (b)**. This means you must open records for **cell 7**, **cell 8**, and **cell 9**, and **cell 26**.

#### 3.6.2.1. Using timing and other tricks to figure out connectivity

For all the cells, zoom into the records so that the x-axis displays activity from 0 to 50 msec, then hit **Init & Run**. Your windows will look like those below.



The first thing you will notice is that all four cells are firing. However, they fire for different reasons. Since similar sorts of things happen in Swimmy's swimming circuit, the following discussion should be helpful preparation for studying that circuit.

First of all, if we had not already told you how the cells were interconnected, you could make some educated guesses about this by looking at their timing relationships. Notice that all of the cells fire regularly at the same rate. On closer inspection, you will see that it looks like the cells fire in a regular sequence such that **cell 7** fires first, then **cell 8** and **cell 26** do so together, then finally **cell 9** follows. This is consistent with the given circuit, **circuit (b)**, in which **cell 7** excites both **cell 8** and **cell 26**, and **cell 8** excites **cell 9**.

Try measuring the time between the *peak* of a spike in **cell 7** and the *start* of the corresponding EPSP in **cell 8**. Since you know that the two cells are monosynaptically connected, the delay should be about 1 msec. Make sure that this is true. The delay between the spiking in **cell 7** and **cell 8** is partly due to the synaptic delay and partly due to the time it takes for the EPSP in **cell 8** to reach its critical firing level. Together, these two factors make the spikes in the two cells occur about 2 msec apart.

If you didn't already know the circuit, the timing relationships would provide evidence for one of the following three possibilities:

- (1) **cell 7** directly excites both **cell 8** and **cell 26**
- (2) **cell 8** or **cell 26** directly excite **cell 9**
- (3) both **cell 8** and **cell 26** directly excite **cell 9**

In each of these scenarios, **cell 9** would have an EPSP beginning 1 msec after spikes in **cell 8** and **cell 26**.

To determine which cell(s) excite cell 9, briefly hyperpolarize of the potential presynaptic cells, **cell 8** or **cell 26**, and see what results in **cell 9**. Conduct this experiment to see for yourself. Are the results of this experiment consistent with the assertion that **cell 9** gets excitation from **cell 8** but not from **cell 26**? Be prepared to address this in **question 3** of your lab report.

### 3.6.2.2 Spontaneous activity

All four cells in **circuit (b)** generate a train action potentials, but not for the same reasons. Cells can generate action potentials either because they get excitatory input from other cells or because they are inherently (“endogenously” or “spontaneously”) active. There are cells of both sorts amongst the **circuit (b)** cells.

**Cell 7** is endogenously active. It fires because its resting potential is above its critical firing level (threshold). The remaining cells, on the other hand, have their resting levels below their critical firing levels and thus only fire when they get input from other cells. To prove this, first hyperpolarize **cell 8** sufficiently so that it does not fire. You will unmask EPSPs in this cell that follow spikes in **cell 7** by one synaptic delay (1 msec in Swimmy). This tells you that although **cell 8** does not fire, it still receives input from **cell 7**. You will also notice that **cell 9** exhibits no activity (it displays a flat baseline), indicating that it too is reliant on input from the presynaptic cell. Because **cell 8** is not able to fire, it is not able to create EPSPs in **cell 9**. To carry out this experiment, open a **cell 8 stim** window and arrange to stimulate the cell, at a negative amplitude, for the duration of the record (from 0 to 50 msec). You may have to zoom into your records to better visualize the effects of this stimulation.

Secondly, if you remove the **cell 8** hyperpolarization (zero out the amplitude in the **cell 8 stim** window) and hyperpolarize only **cell 7** so that it no longer fires, you will observe a flat baseline in **cell 7**, indicating that its firing was not caused by input from some other cell. You will also see that without input from **cell 7**, activity in all the other cells disappears, indicating that their activity was directly or indirectly dependent on **cell 7**.

We could describe the situation in this circuit by saying that **cell 7** is the origin or source or *generator* of all spontaneous activity and that the other cells are *followers* of **cell 7** with respect to their activity. Such a neat distinction between generators and followers is not always possible, but trying to make this distinction can sometimes help to make sense out of a complicated circuit.

You may now close the records of all the cells in **circuit (b)**, since they have no involvement in the remainder of this project.

## 4. Week 1 Homework

**The following sections are necessary to complete the quiz (due at the start of lecture 2) and the assignments (due at the start of lab 2).** You will need to read about theoretical ideas regarding the mechanism behind generating rhythmic bursting, otherwise you will not be able to effectively do the work needed in the next lab sessions. Read and understand the following material and then complete the take-home quiz handout and the assignments detailed below.

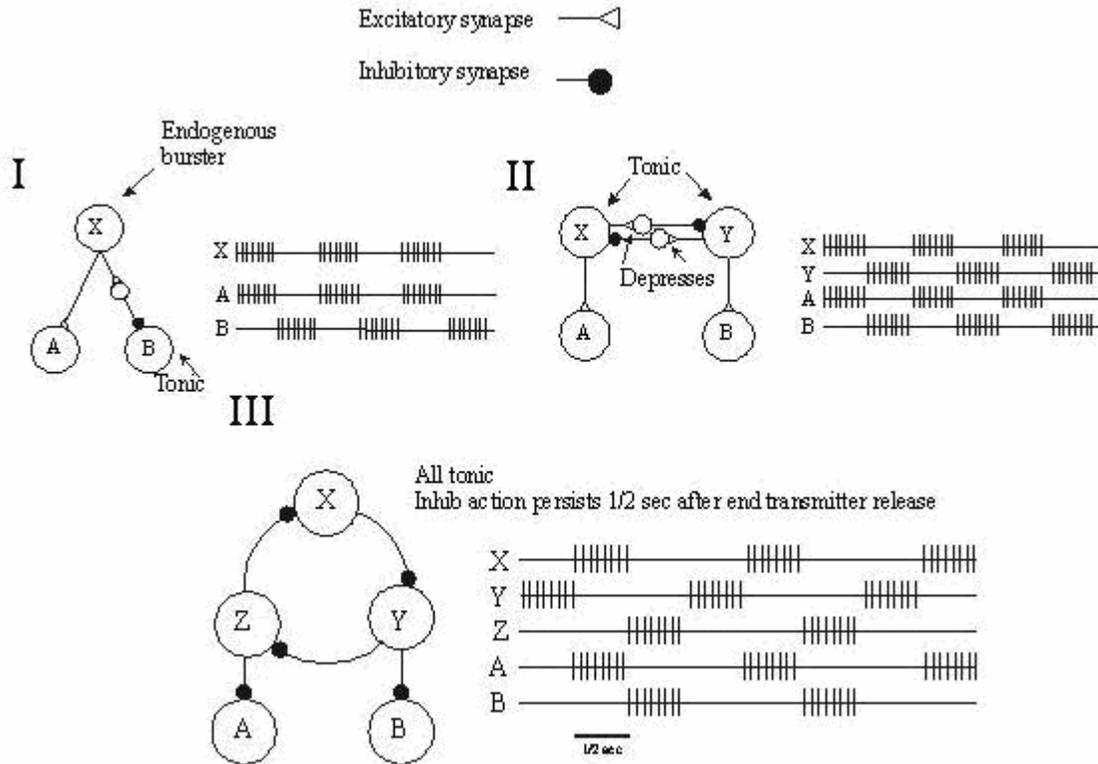
### 4.1. Theoretical background (NOTE: This material will be the subject of a quiz)

A common misconception has scientists collecting data on a phenomenon or topic, and then, based on those data, arriving at an explanatory hypothesis, which they then test with further experimentation. In reality, scientists come to study of particular phenomena based on hypotheses arising from ideas about whether or not a particular phenomenon *could* work or from preconceived notions of how nature *ought* to work. Rarely does a hypothesis come to mind simply by looking at data; it is much more common, given the limitations of human intelligence, to start somewhere and refine. As you will probably find out for yourselves in this module, it is much easier to test ideas about how something might work than to generate the ideas from raw observations.

Locomotion typically involves rhythmic movements. The tail of a fish, for example, moves first left, then right, in continuous alternation. The legs of a biped step forward in rhythmically as flexor and extensor activity in a given leg alternates. Usually, each movement of a locomotor sequence is due to the contraction of a muscle that occurs because motor neurons to that muscle fire bursts of spikes, which is something you saw with Swimmy’s motor neurons.

Let us suppose, for the sake of discussion, that there are exactly two muscles, labeled **A** and **B**, which fire in rhythmic alternation. Each of the three following schemes exemplifies a theory that has been proposed and seriously considered by neurophysiologists to explain the rhythmic pattern in these two muscles. The circuits that you are about to study in

class will incorporate similar principles to those discussed here, but none of the circuits will be an exact copy of one of the schemes below.



- (I) The **endogenous burster hypothesis** proposes that rhythmic movements of locomotion originate from cells whose intrinsic membrane properties cause them to produce rhythmic bursts of spikes. (It is not necessary to focus on why certain cells exhibit these endogenous properties, since that would go beyond the scope of this class.)

In figure I, X sends bursts of signals to A, causing A to mimic the firing pattern in X. In order to alternate rhythm, B must fire only when A is not. One way to arrange this is to send inhibitory signals to an endogenously tonically active B at the same time that A is receiving excitatory ones. The inhibitory interneuron, shown above, and A are given simultaneous excitatory bursts of activity. The interneuron then inhibits B so that its tonic activity is halted around the time that A is firing. (Note that endogenously tonic activity results when the resting potential of a cell is higher than the critical firing level, and is fairly common in nerve cells.)

- (II) The **mutually depressing inhibition hypothesis** proposes that rhythmic movements begin with endogenously tonically active cells that mutually inhibit one another via synapses that are prone to depress with use, and recover with rest.

In figure II, tonic cells X and Y send excitatory signals that weaken over time (they undergo depression) to inhibitory interneurons. The interneurons relay these weakening signals through inhibition of the opposing cell. For example, as the tonic activity of X tires, the opposing cell, Y, becomes less inhibited by the interneuron and eventually is able to fire while X rests and recovers activity. Thus an alternating pattern of bursts occurs in X and Y, which directly initiate the same in A and B.

- (III) The **ring oscillator hypothesis** proposes that rhythmic movements are due to endogenously tonically active inhibitory neurons whose effectiveness continues for a certain amount of time (for example 1/2 sec) after it ceases to fire because its transmitters are slow to be inactivated.

In figure III, tonic cells **X**, **Y**, and **Z** form a ring of inhibitory signals. While one of these cells fires, the postsynaptic neuron is inhibited both during the firing and for 1/2 sec after this. This pattern cycles about the neurons in the ring. If **A** and **B** are also tonically active neurons, they will cease to fire while the **Z** and **Y** are in effect.

There are several more hypotheses that have been advanced, but this will suffice for our purposes.

**Your task:**

Complete the Quiz handed out by your instructor. This will be due at the beginning of the next lecture.

#### 4.2. Finding neurons involved in swimming

Now that you have become familiar with the Swimmy program and have practiced analyzing and manipulating simple circuits, you may begin working with Swimmy's neural circuit. This is the circuit responsible for Swimmy's rhythmic locomotive activity.

Swimmy has 26 neurons. You are already familiar with the left and right motor neurons (**cell 1** and **cell 2**, respectively) and the simple circuit cells (**cells 3-9** and **26**) that you worked with in **Week 1 Lab**. Of the remaining neurons (**cells 10-25**), only some are involved in swimming and others are not. A good place to begin determining which neurons are involved in swimming is to look for neurons that produce a similar pattern of activity as seen in the motor neurons. Cells will display bursts of spikes with similar periodicity as the swimming motor neurons, occurring either in phase or out of phase with one of them.

Conceivably some cells involved in the swimming circuit would not exhibit this sort of pattern, and therefore it would be more difficult to conclude that they belong to the circuit. It is best to assume that there are no such neurons, and the circuit you will be using only has cells that display a similar pattern to either **cell 1** or **cell 2**.

**Your task:**

Find all the cells involved in the swimming behavior and make a list of them to hand in. To do this, simply search through the records for **cells 10-25** and compare their activities to those of **cell 1** and **cell 2**. (Note: You might want to use the **8 Windows** feature to speed this along.) Do not copy someone else's list! This experience will help you progress in this module.

#### 4.3. Learning from just one cell at a time

Fortunately for you, Swimmy has far fewer neurons than any real animal. In a real animal, the task of finding all the neurons involved in any given behavior can take years! Once such a neuron is found, it should be used to explore as many questions of interest as possible, particularly because the same neuron might never be found again. In various lower animals, some neurons are recognizable by their appearance under the microscope and can be recorded from at will, as can be done with Swimmy's neurons (though not as easily as with Swimmy). Typically, researchers only know of a general region in which to search for neurons, and an electrode must find its way into a specific neuron by chance. When this happens, the neuron becomes subject to various kinds of stimuli and other tests, and is classified based on its responses.

When studying rhythmic locomotor movements, researchers primarily wish to determine the origin of the spontaneous rhythm. Cells involved in the circuit may be responsible for directly generating the rhythm ("leaders" or "generators"), or they may simply respond to such generative cells ("followers"). One way to determine if a cell is a generator is to temporarily manipulate it, usually by hyperpolarizing the cell so as to stop firing for a while or by briefly depolarizing

the cell to make it fire where no prior action potentials exist, and observe any changes in its subsequent firing pattern. A phase shift indicates that the cell is a generator, since a phase shift requires that the cell not be under control of another cell's firing pattern. An unaltered pattern, however, suggests the cell may just be a follower.

Upon discovery of a generative cell, it is important to determine what type of oscillator it belongs to (endogenous burster, mutually depressing inhibition, or ring; **section 4.1**) by questioning how it functions with other cells to contribute to the rhythmic oscillation. For example, if you hyperpolarized a generator in a mutually depressing inhibition oscillator, what pattern of input would you see in the other cell(s) in the generator circuit?

Because Swimmy has so few neurons, you are instructed to *first* find all the neurons involved in swimming and determine the connections between them, and *then* find the neurons that are part of the generation circuit and determine the mechanism of oscillation. Note that this would not be an efficient order when studying real animals.

**Your task:**

Pretend, for this section, that Swimmy is a real animal and you are formally researching its circuit. Experiment with and observe **ONLY ONE CELL AT A TIME** and determine if it may be a generator or a follower. Study each of Swimmy's cells until you find one that you believe is a generator, then study that cell further to determine which of the three possible oscillating mechanisms (**section 4.1**) apply to the circuit. If your instructor is so inclined, he may suggest a limited number of cells for you to study, and provide you with additional guidance in this task.

#### 4.4. Beginning to work out the swimming circuitry

Once you have a list of neurons that seem to be involved in swimming, you can begin to figure out how they are interconnected and what makes them produce action potentials. The best strategy for getting started is to use the fact that synaptic delays in Swimmy are about 1 msec long. In direct synaptic connections, spikes are followed by the onset of an EPSP or an IPSP in another cell just 1 msec later. You can hypothesize that such cells are directly connected. For example, if **cell m** directly inhibits **cell n**, then the peak of every spike in **cell m** should be followed by the start of an IPSP in **cell n** almost exactly 1 msec later.

Once you have made hypotheses regarding the swimming circuitry, you may conduct stimulation experiments to confirm or deny the existence of a given connection.

**Your task:**

Use the above approach to determine the neuron(s) that directly:

- (1) excite cell 1
- (2) inhibit cell 1
- (3) excite cell 2
- (4) inhibit cell 2

Make a list of these results to hand in with the list you made for **section 4.2**. Again, this will be due at the beginning of the next lab.

### 5. Week 2 Lab

By now you should know all the cells that are involved in your swimming circuit (including **cell 1** and **cell 2**) and you should have identified the exciters and inhibitors of **cell 1** and **cell 2**. Use your lab period to determine the remaining direct connections in the circuit, the cells that generate the rhythmic pattern, and the mechanism these generators employ to create this pattern. You will be instructed on how to provide graphical evidence for these topics. The following helpful helpful hints will aid you in this process.

- (1) Swimmy's swimming circuit is bilaterally symmetrical. Therefore, if a connection exists on one side of the nervous system, there will be a homologous connection on the other side as well.
- (2) **SILENCE IS GOLDEN!!!**
- (3) Follow the Principle of Parsimony, which avers that the simplest logical solution should be considered the correct one, unless there is a good reason for doing otherwise.

Once all of the connections have been determined and announced in the lab, you will still have several remaining tasks. One is to obtain experimental evidence for each of the connections that have been announced, since you will have to provide these in your lab reports. Because the circuit is bilaterally symmetrical, you need only provide evidence for one side of connections. Another task is to determine how the circuit functions. Be prepared to address the following questions when analyzing your circuit and writing your lab report.

- (1) Which neurons are responsible for generating the rhythm and which are simply followers whose activity has no influence on the pattern?
- (2) How is the rhythm generated? Which of the theories described in **section 4.1**, if any, is responsible for this rhythm?
- (3) How does the circuit work over-all? What are the roles of all the neurons, including the follower neurons, and why are their connections important?

Questions (1) and (2) will be answered at least in part on the basis of experiments. Question (3), like any question of biological function, is a matter of interpretation and may depend more on logic than on experiments.

## 6. Lab Report

YOUR LAB REPORTS MUST BE ENTIRELY YOUR OWN, PERSONAL, INDIVIDUAL WORK. Your grades will be based on the following criteria:

- (1) accuracy of conclusions
- (2) credibility of arguments and evidence
- (3) clarity of writing and presentation of data

**\*\*\* DO NOT WORK WITH OTHERS!!! \*\*\***

Please organize your report *according to the numbers of the questions below* to make it easier for instructors to grade. You should indicate at the start of each section which question you are addressing, but do not actually repeat the questions themselves.

All of your printouts of experimental evidence (those containing records, and **cell stim** windows) should appear immediately after the question and NOT at the end of the report. All graphs included in your report must come from experiments done by you and **must include your label (the window containing your station name) in the upper right corner**. On rare occasions students have had problems getting their **station name** to print. If this should happen either start the program over and try again or get an instructor to initial your printouts.

When writing your lab report, be sure to label the top of each printout with the connection to which it pertains (e.g. "cell 6 inhibits cell 4"). Also, for each printout indicate its status: *Is it a demonstration of synaptic delay? Is it a baseline/control result for comparison to some experimental procedure? Is it an experimental procedure? Etc.* For synaptic delay printouts, draw arrows to indicate where the synaptic delay begins (in the presynaptic cell) and ends (in the postsynaptic cell). Labeling of printouts can be done in pen.

**NOTE:** *You must have actually done the experiments detailed in your lab; it will be considered CHEATING to provide copies of graphs made by a different person.*

### Swimmy Lab Report Questions:

1. Refer to **section 3.6.1.4** on inhibition, **section 3.6.1.5** on effects of altering membrane potential on IPSPs, and **section 3.6.1.6** on effects of shunting on membrane potential. [You do not need to provide printouts as evidence for this question.]
  - a. [**section 3.6.1.4**] If one stimulates an excitatory input during inhibitory action, the excitatory input produces an EPSP that is less likely to exceed the neurons critical firing level than in the absence of the inhibition. Is this because the EPSP itself is reduced, because the IPSP subtracts from it, or both?
  - b. [**section 3.6.1.5**] What happens to an IPSP as the resting level becomes more and more hyperpolarized?
  - c. [**section 3.6.1.6**] What happens to an EPSP if it occurs during the time of inhibitory action where there is no visible IPSP (because the membrane is already at the inhibitory reversal potential)?
  - d. Explain the mechanisms for each of the effects described above.
2. Refer to **section 3.6.1.7** on short-term plasticity. Do the excitatory synaptic inputs from **cell 3** to **cell 4** and from **cell 5** to **cell 4** facilitate or depress? Show figures as supporting evidence for your answer. Why is it important to separate the two events in time in order to establish facilitation or depression and rule-out other explanations? Clearly explain the difference between facilitation and summation (and how they could be confused). Clearly explain the difference between depression and shunting (and how they could be confused). Finally explain the difference between depression and inhibition.

3. Refer to **section 3.6.2.1** on synaptic delay. Explain how you would conduct an experiment to show the following:
  - a. **cell 26** does NOT drive/excite **cell 9**
  - b. **cell 8** drives the activity in **cell 9**
  - c. **cell 7** is the generator and the rest of the cells are followers

Provide printouts to support your answer.

4. Draw the circuitry in which the swimming cells participate, using the same drawing conventions shown in lab, and be sure to give the correct number of each cell. Indicate any cells that have endogenous properties (those that endogenously burst or those that are endogenously tonic). Indicate any synapses that facilitate or depress only if it is important to the oscillating mechanism of the circuit. [No evidence required in this part of the report.]

5. Cite evidence for each of the direct synaptic connections in your circuit. You only need to show experiments for one side (left or right) of Swimmy, since the circuit is bilaterally symmetrical. We expect that for each connection you will have established that it is monosynaptic by determining synaptic delays and that the presynaptic cell is causing the EPSP or IPSP in the postsynaptic cell. Please include your printouts establishing both these points. For each such experiment provide a brief description of what you did and explain what the data proves.

6. Which neuron(s) are essential for generating rhythmic bursts of firing (which cells are pattern generators, as opposed to being mere followers of other cells)? Briefly discuss the experiment you performed, and how it leads you to your conclusion. Show evidence (printouts) that the cell(s) generating the pattern are both necessary and sufficient for producing the rhythm. Explain why you conclude that the remaining cell(s) are NOT generating the rhythmic bursts. [You are not responsible for providing evidence that these cells are followers.]

7. Having identified the generators, describe the mechanism that these neurons use to generate bursts. Provide graphs of crucial experiments that establish the oscillating mechanism. Also explain how your evidence disallows the conclusion that the rhythm of the circuit is produced by other types of oscillators. If the bursting rhythm is being generated in a manner similar to one of the theories described in the lab manual (**section 4.1**), do not just refer to that theory, but actually explain its mechanism so we can see that you understand it.

8. Explain how the circuit, *as a whole* and not just the neurons involved in generating the rhythm, works. (DO NOT just reiterate the types of connections, although this could be part of the story.) Briefly reference each set of connections and why they are necessary to the circuit. Walk the reader through the circuit and describe what happens at each stage. Perhaps it is best to think of Swimmy as layers of neurons and describe what each layer is doing in the circuit. [No evidence is required; only a clear description is needed.]

9. Now that you know how the circuit works, you should be able to understand why malfunctions of the circuit cause the behavioral abnormalities that they do. Stop the rhythmic pattern generator neuron(s) on one side of the circuit and see what this does to the behavior. Describe the resulting behavior and explain it in terms of your understanding of the circuit mechanism. Contrast these results with your observations from **section 3.5**, where you examined the activity of Swimmy's tail while one of its motor neurons was hyperpolarized.

10. Suppose you could construct two different circuits both of which could explain *all* your data equally well but one of which was more complicated than the other. How would you choose between them, or would you?

# SWIMMY SURVIVAL GUIDE

Summary of how to do most things needed when running NEURON/Swimmy.

## Starting and closing Swimmy

To start NEURON go to **SwimmyGUI** folder (using **Windows Explorer**), and double-click the **StartSWIMMY24** batch file. Type in the name of the Swimmy program you were assigned, then press **Return** on the keyboard. The six possible versions are listed below; circle the one you will be using for your own reference.

swmA24  
swmB24  
swmC24  
swmD24  
swmE24  
swmF24

To quit Swimmy click on the **x** box in the upper right hand corner of the **nrnalone Command Window**.

## Generalities

**Left- and right- clicking.** Unless otherwise specified, left-click; however, some specified operations require right-clicking, and these will be indicated as appropriate in the tutorial.

**Entering information in boxes.** In order to enter information into control windows boxes when running NEURON programs, the mouse pointer must *be* in and *remain* in the space allotted for typing. If it moves away you will have to put it back if you want to continue what you were writing in the window.

**Case sensitivity.** NEURON is very case-sensitive. File names are not case-sensitive, but most everything else is.

**Moving, sizing, and closing windows.** To move windows drag with your cursor on the blue strip at the top of the window. If you try to drag from somewhere else, you may end up inadvertently closing the window. You resize windows in the normal manner, but if you try to make them too small some information at the edges may be cut off or distorted. NEVER use the **resize** option in the **Print & File Window Manager**. To close windows click on **Close** in the gray bar.

**Slashes.** Whereas Microsoft Windows uses backward slashes ( \ ) to separate items of paths, NEURON uses forward slashes ( / ). You may never have to write a path in a NEURON window; however, if you do, use forward slashes.

## Graphing membrane voltage of Swimmy neurons

**Creating individual graphs.** In the **Swimmy Stuff** window select **Make Record**. Select the cell number you wish displayed from the drop menu. This will cause a graph window to appear, with the appropriate cell name on it.

**Creating eight graphs at once.** In the **Swimmy Stuff** window select **8 Windows**. A new window labeled **8 Windows** will appear. Determine which 8 windows you wish to view, place the number '1' in the corresponding eight boxes, and click **MAKE GRAPHS**.

**Changing the scale of a graph.** Right-click on the graph and select **View/Set view**. Put new values in white window as necessary.

**Removing graphs.** To get rid of a graph click on **Close** in the horizontal gray bar near the top of each window.

**Changing what is plotted in an existing graph.** Sometimes you may want to look at a lot of different cells, one at a time (for example, when surveying cells to see which ones might be involved in swimming). Rather than making a new graph for each, you can right-click on a graph and select **Change text**. Once you have done that, clicking on the line in your graph that says what is being plotted will open a little window in which you can edit what is there and thereby change it.

**Superimposing traces.** If you want to keep previous traces on a graph when new ones are added, so as to facilitate comparisons, you may right-click on the graph and select **Keep Lines**. You may then clear the graph window by right-clicking and selecting **Erase**. Uncheck **Keep Lines** when you are finished with this option.

**Plotting more than one thing on a given record.** You may plot as many different things (cells) on the same graph as you want. Simply create additional **Plot What** operations (please consult the instructor for more details). In order to distinguish the different lines, you will want to plot different variables in different colors. To do this, right-click on your graph window and select **Colors**. Then choose the color you want and click on the label of the variable you want to plot in a color. Sometimes, after having arranged to color the graphs of some variables, you will find that when you go back and try to change to new colors, you will not be able to get the color menu. If this happens, click once on **delete**; then you will be able to get the color menu again.

**Deleting lines or variables from a record.** To delete selected lines from a graph or selected variables, right-click on the relevant graph and select **delete**. Then left-click on whatever items you want to remove.

### Establishing intracellular current passing electrodes

Select **Make Stim** in the **Swimmy Stuff** window, and choose the cell you wish to manipulate. This will bring up a window called **cell n stim**, where **n** is the cell number you selected from the drop window. Use the delay, duration, and amplitude buttons to create the desired stimulation. To get rid of an electrode, set the **amplitude** to **0**, hit **Init & Run** to make sure you have eliminated it, and hit **Close**.

### Recovering hidden items

It is easy to accidentally hide windows in Swimmy. To recover the **Run** or **RunControl** window, simply select **Tools** in the **NEURON Main Menu**, then click on either **RunButton** or **RunControl**. The corresponding window will appear.

To recover a hidden **cell stim** window, select **Tools** in the **NEURON Main Menu**, then **Point Processes**, then **Viewers**, then **Point Processes**, then **IClamp**. Double-click the item corresponding to a **cell stim** window to view that window.

To recover other hidden windows, select **Window** in the **NEURON Main Menu**, then click on any items in the list that do not have check marks next to them.

### Printing

**Printing graphs internally (using Swimmy).** In preparation for printing you should make sure the windows you want to print are not more than about half the screen wide; otherwise they won't fit on the printed page. Then click **Window** in the **NEURON Main Menu** and select **Print & File Window Manager**. This will bring up a window with two red rectangles. The one on the left will be a map of the windows you have open. Click on **select** and then click the windows you want to print. They will be reproduced in or around the right red box, which is a rough representation of the page that will be printed out. You can move the rectangles in this box around on the page as you like if you first click **move**. Once you have placed the windows you want in the page rectangle, you can make further adjustments to their size by changing the size of the original graph, and it will also change in the page rectangle.

When you have everything as you want it, click **Print** and then **To Printer**. You may get an additional window with an **Accept** button; if you do, then click on that as well (if you get a blank sheet of paper printed, which sometimes happens, just print the page again; the second time it always seems to work).

**Printing graphs externally (using screenshots).** If you are having trouble printing using Swimmy's print option, you can instead print either a single screenshot of the entire computer screen or a collection of screenshots of individual windows.

To capture and print everything currently visible on your computer desktop at once, press the **PrintScrn** button on your keyboard to copy the entire screen, paste the image into a word processing program (like Microsoft Word or OpenOffice SWriter), and print out the document. This is the quickest way to print externally.

To capture and print individually selected windows from your computer desktop, select a window (by clicking on it once), press both **Alt** and **PrintScrn** simultaneously on your keyboard to copy that window, paste the image into a word processing program, repeat until you have gathered all of the windows you wish to capture, and print out the document. This will produce the cleanest-looking screen images when printing externally.

### **Name identification**

In order to distinguish your graphs from those of other groups, you must include a window containing your name. To do this, click the **Station** button on the **Swimmy Stuff** window. Enter your name in the **NEURON** window that pops up and click **Accept**. By including this little label window on all of your printed pages you will be able to identify the graphs belonging to you that are being printed out on the lab's printer.

# DOWNLOADING AND RUNNING SWIMMY GUIDE

Summary of how to ensure that Swimmy is properly downloaded and functioning.

## Preparing your computer for the SwimmyGUI download

Before downloading **SwimmyGUI**, make sure your computer has the following components.

### (1) **Microsoft .NET Framework Version 1.1 Redistributable Package**

This is required in order for the Swimmy animation to run, and without it the **SwimmyGUI** program may not even unzip. To acquire this component, download **dotnetfx.exe** from the following website.

<http://www.microsoft.com/downloads/details.aspx?FamilyID=262d25e3-f589-4842-8157-034d1e7cf3a3&displaylang=en>

You can easily find this download page by searching **dotnetfx.exe** in a search engine, rather than typing in the entire link.

### (2) **C:**

You also need to ensure that your computer has a **C:** drive. This will allow Swimmy to initiate its animation window.

## Downloading SwimmyGUI to a Windows-equipped computer

To download the full **SwimmyGUI** program from the website, click once on the link shown below.



▪ [Download SwimmyGUI-v24.exe - \(3.54 MB\)](#)

Opt to save the file by clicking the appropriate button.



The file should automatically save to your computer's desktop. If you cannot find the file there, search for

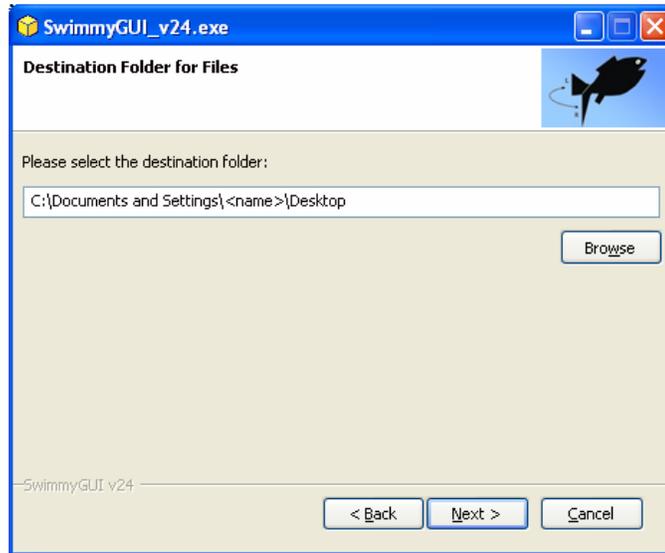
**SwimmyGUI-v24** on your computer. If you still cannot find it, try saving the file again. Then double-click the **SwimmyGUI-v24.exe** icon. Choose to run the program if you are given the option of doing so or not.



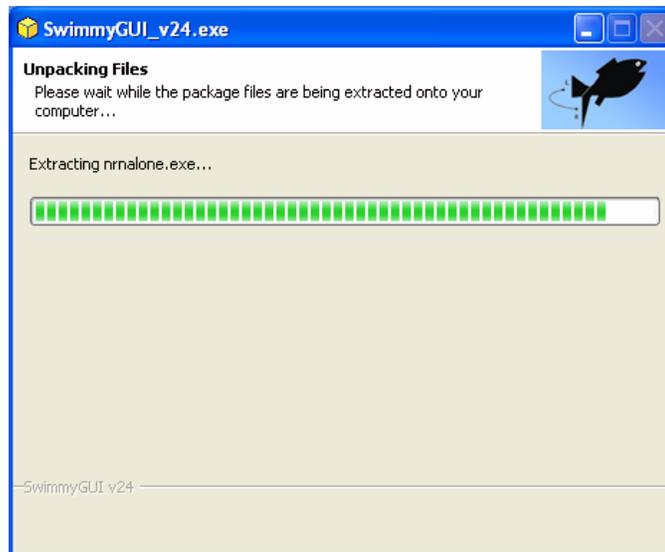
A **Welcome** window will appear.



Click **Next >** on the **Welcome** window. A **Destination Folder for Files** window will appear.



The destination folder will default to the computer **Desktop**. If you would rather the **SwimmyGUI** program extract and save to another location, simply browse for the desired location. Then select **Next >** to proceed with the download. An **Unpacking Files** window will appear only while the extraction is occurring and will automatically close when finished.



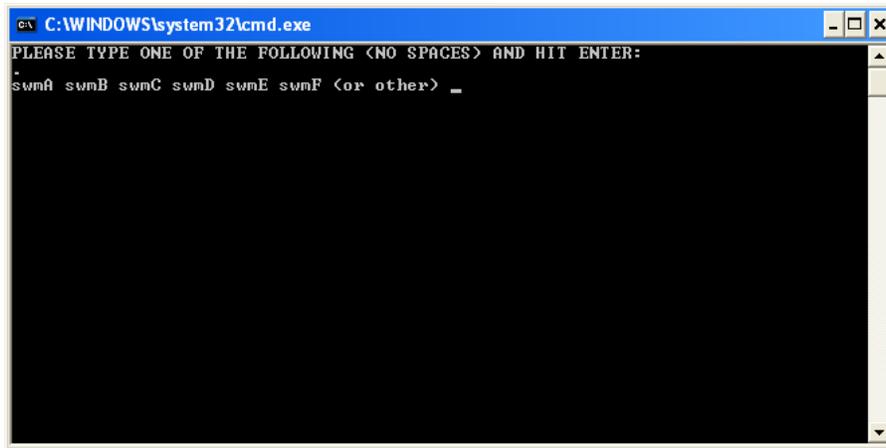
Check to see if the extracted **SwimmyGUI** folder can be found on the **Desktop** (or wherever else you specified).



Follow the instructions given in your **SwimmyGUI** tutorial beginning at **section 3**, to run the program.

### Starting Swimmy by typing in the program name (standard method)

This is the most convenient way to start Swimmy. To begin using the Swimmy program, go to the **SwimmyGUI** folder and double-click the **StartSWIMMY24** batch file.



A window will appear listing the six different student versions of Swimmy. Circle the one you will be using for your own reference.

swmA  
swmB  
swmC  
swmD  
swmE  
swmF

Type the program name in the window, then hit **Return** on the keyboard to load the program.

### Starting Swimmy using a command prompt (alternate method)

If you are having trouble starting Swimmy by double-clicking the **StartSWIMMY24** batch file and typing in your Swimmy program name (which might be the case if a version of NEURON has previously been installed on your machine), you can do it a more cumbersome but more reliable way using the **nrnalone** command window.

Open the **SwimmyGUI** folder, followed by the **nrnpgms** folder. Double-click the **nrnalone.exe** icon. A **nrnalone** command window will open.

```

nrnalone
File Help
NEURON -- Version 5.2 2002/03/28
by John W. Moore, Michael Hines, and Ted Carnevale
Duke and Yale University -- Copyright 2001

Loading membrane mechanisms from nrnmec.dll
Additional mechanisms from files
CHAYHH.MOD EPlasInp.mod EPlasSom.mod EPlasSomf.mod EPlasSyn.mod EPlasSynf.mod ExIAF.mod FHH.MOD HHALT.MOD HHCA.MOD HHCA1.MOD I
NPUT.MOD InhIAF.mod PACEF.MOD
With CCode
"jvmdll" not defined in nrn.def
JNI_CreateJavaVM returned -1
Info: optional feature Java VM is not present.
oc>

```

At the prompt (**oc>**) type the the following command EXACTLY as written below. Enter your **Swimmy** program name, followed by the version number, **24**, followed by the extension, **.hoc**, where it says *program*.

```
load_file("${NEURONHOME}/nrnpgms/program")
```

Therefore, you should have entered one of the following commands.

```

load_file("${NEURONHOME}/nrnpgms/swmA24.hoc")
load_file("${NEURONHOME}/nrnpgms/swmB24.hoc")
load_file("${NEURONHOME}/nrnpgms/swmC24.hoc")
load_file("${NEURONHOME}/nrnpgms/swmD24.hoc")
load_file("${NEURONHOME}/nrnpgms/swmE24.hoc")
load_file("${NEURONHOME}/nrnpgms/swmF24.hoc")

```

Hit **Enter** and your program should start.

Since this method does not initiate Swimmy's animation window, you will need to open the window yourself. See the instructions below to achieve this.

### Initiating the Swimmy Animation window

Swimmy comes equipped with an animation window through which you can visualize the movement of your virtual fish. If you are having trouble starting Swimmy by double-clicking the **StartSWIMMY24** batch file and typing in your Swimmy program name, then you will have to start Swimmy using a command prompt (as detailed above) and open the animation window separately (as detailed below).

To access this window, first ensure that your computer has the **Microsoft .NET Framework Version 1.1 Redistributable Package** and a **C:** drive, as detailed in [Preparing your computer for the SwimmyGUI download](#). Next, create a new folder directly within the **C:** drive and name it **Temp**. If you already have such a folder, then you do not need to create one. Then, go to the **SwimmyGUI** folder and double-click the **nrnpgms** subfolder. Double-click the **anim2TempNew.bat** icon (a dark window should flash onto the screen for only a moment and then disappear), then double-click the **RunAnimAtHome.exe** icon. This action will cause the animation window to pop up.