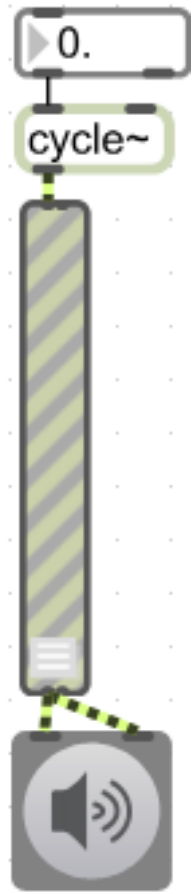
Max/MSP exercises 6a

Ex. 1

Some slightly more complex synthesis, now: AM (Amplitude Modulation) and FM (Frequency Modulation). These work by changing the volume of pitch of an oscillator over time. In both cases you can get interesting results at both low modulation frequencies (which yield tremolo and vibrato respectively) and high modulation frequencies (which yield more complex timbres).

We'll start with Amplitude Modulation



1. Start by copying the following, which will now be familiar to you:

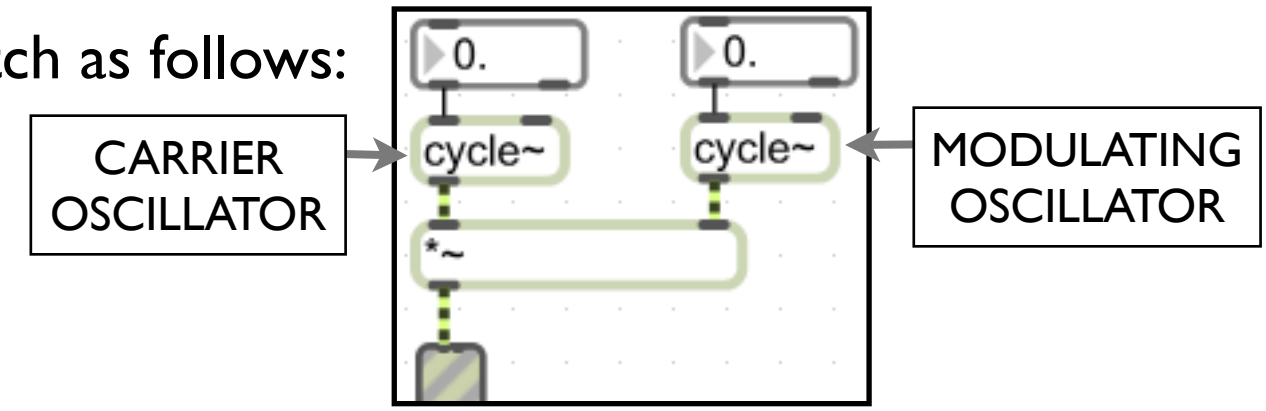
WARNING: remember to keep the level of this [gain~] slider low so that [cycle~]'s output, which will be loud, doesn't damage your ears.

We could say that simply moving the [gain~] fader up and down will result in amplitude modulation since we are changing the amplitude over time. However...,

Ex. 1 (cont)

We need to make the amplitude rise and fall in a consistent way to give us true AM synthesis. For this we can use another oscillator.

2. Modify your patch as follows:



Here we have identical inputs to a [*~] object which controls the amplitude for us (the [gain~] object is just to control output levels). And this will indeed modulate the amplitude, as you will see if you...

3. ...set the left-hand [number] box to 1000, and increase the right-hand [number] box to 2. The amplitude will change over time. Do you notice anything odd about the frequency of modulation? Is it at 2Hz?

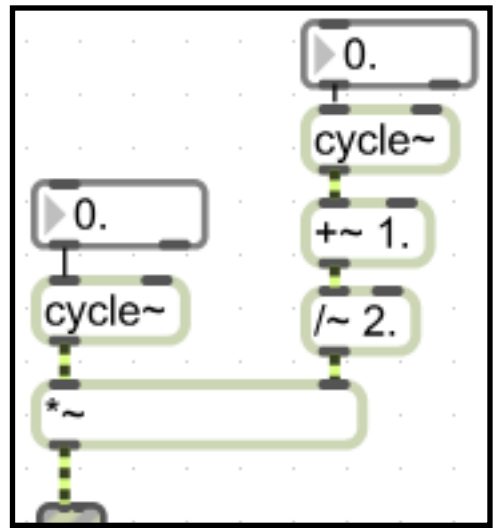
Ex. 1 (cont)

4. Increase the right-hand [number] box to 100Hz gradually. What happens?

You should notice that you get two pitches which diverge. This is a characteristic of Ring Modulation, which in fact is what we have created. The modulating oscillator is in fact increasing the amplitude to 1., then decreasing to 0., then to -1. (i.e. inverting the phase). This is both why

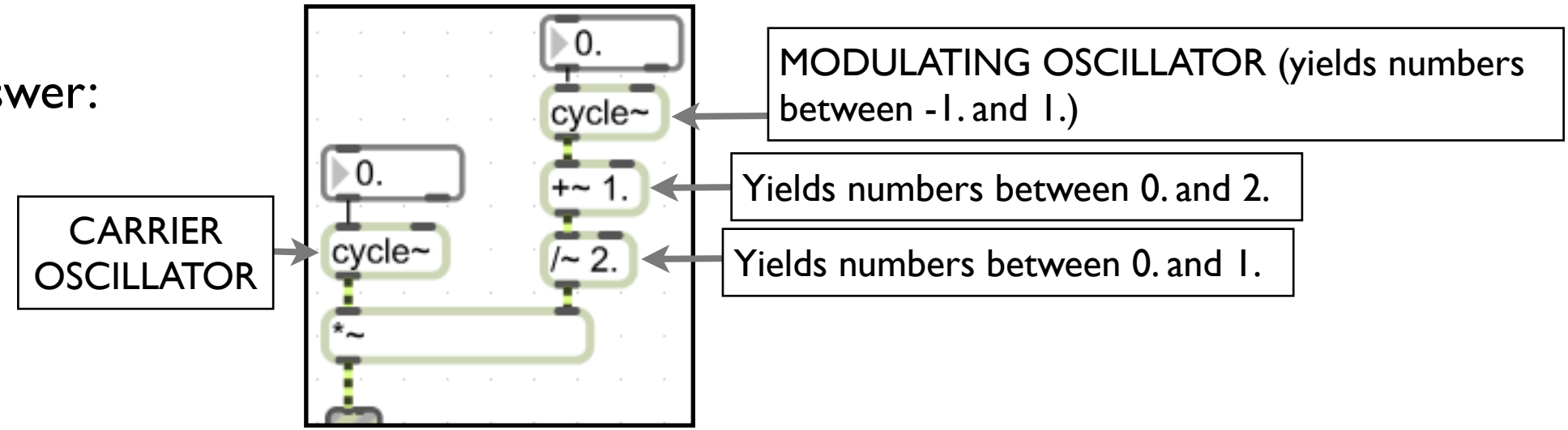
- we hear a 4Hz amplitude modulation when the input to [cycle~] is 2Hz
- we hear two diverging sidebands but not the original frequency, which is cancelled out.

5. In order to ensure that the amplitude does not go to negative values (thus inverting the phase), we need to do a little mathematical jiggery-pokery. Modify the patch as shown. What's happening here?



Ex. 1 (cont)

Answer:



6. Try this out. How are the results different (at both low and high modulation frequencies)?

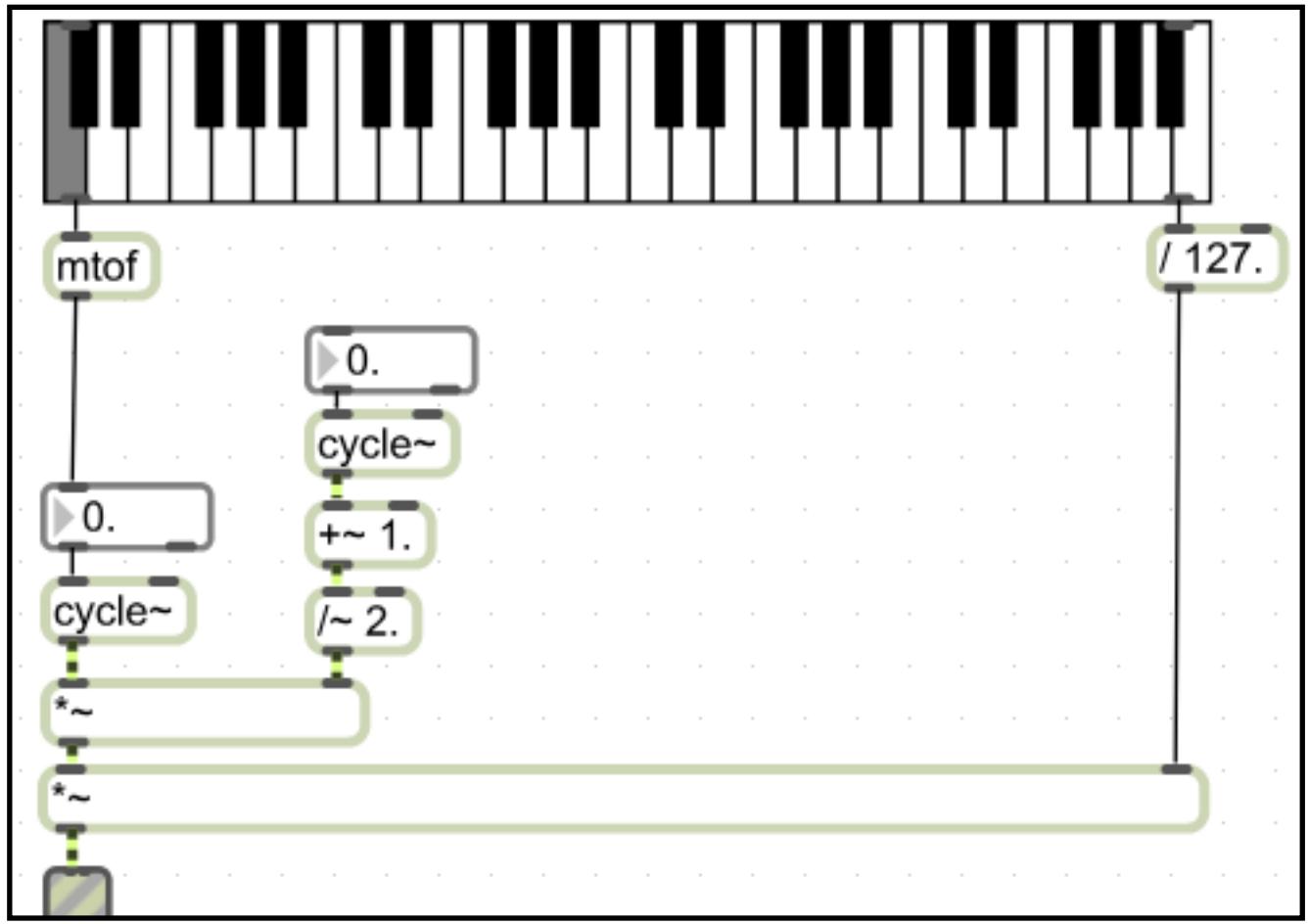
7. Based on what we know from Exercise6a, attempt the following:

• Control the frequency of the carrier oscillator using a [kslider] object or MIDI keyboard

• Control the overall level of output using the right-hand outlet of the [kslider] (you will of course need a further [*~] object to do this.

Ex. 1 (cont)

Solution:



If you try this out you'll notice that the timbre is not consistent over the entire keyboard because the modulating frequency remains the same even though the carrier frequency changes.

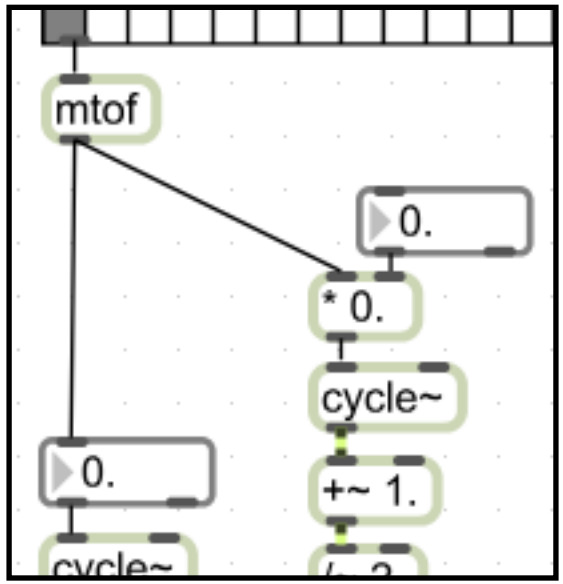
8. Referring to Exercise5b p5-7, do something similiar to:

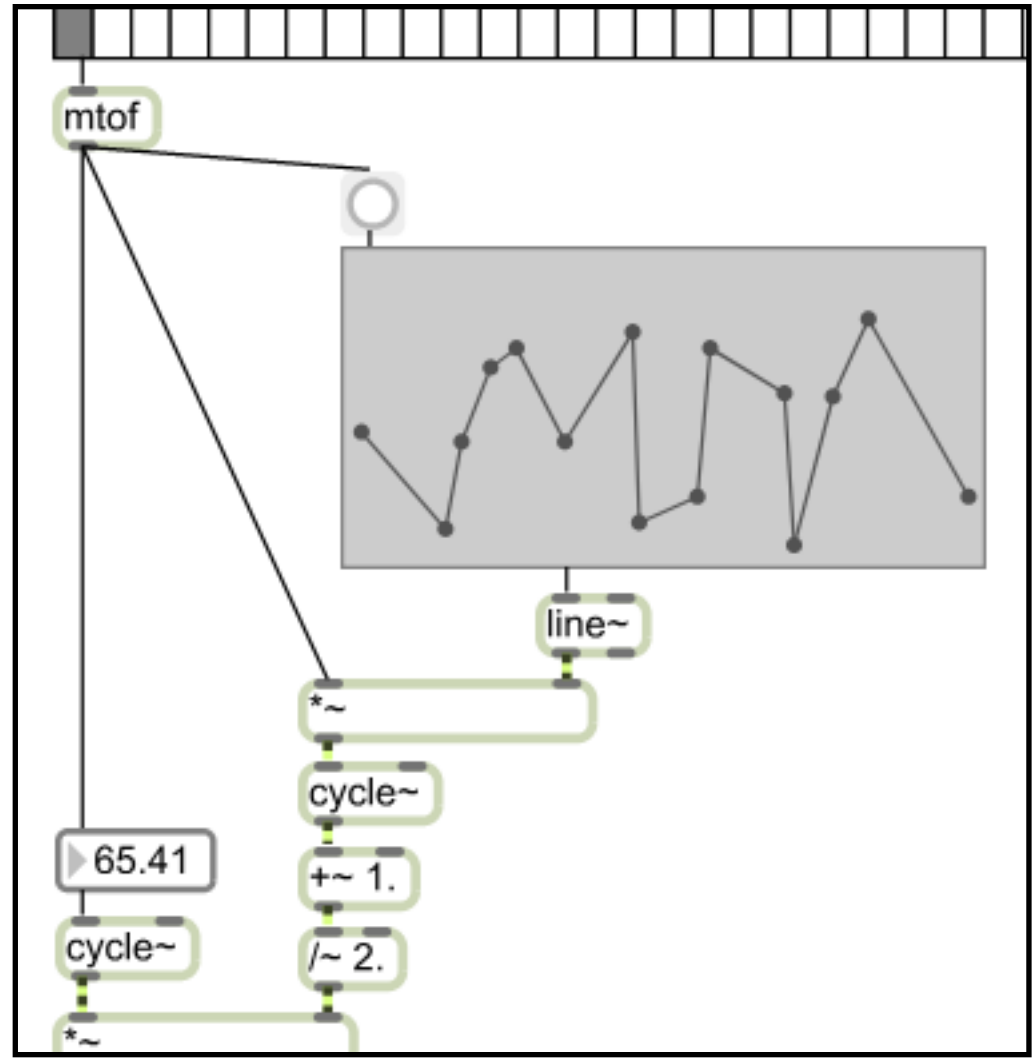
• ensure a consistent relationship between carrier and modulating frequencies;

• permit an evolving timbre by using [line] or [line~]/[function] objects, triggered by the [kslider].

Ex. 1 (cont)

Solutions:

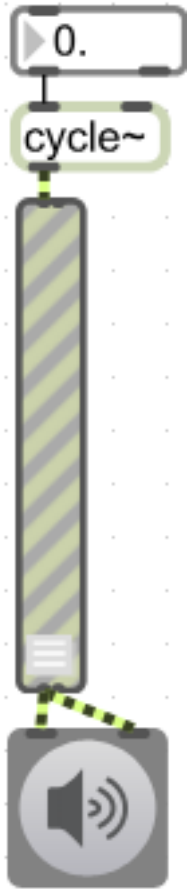




Remember that you will need to change the Lo and Hi Display Range for the [function] object. You might also want to change the Hi Domain Display Value if you want to slow the evolution of the sound.

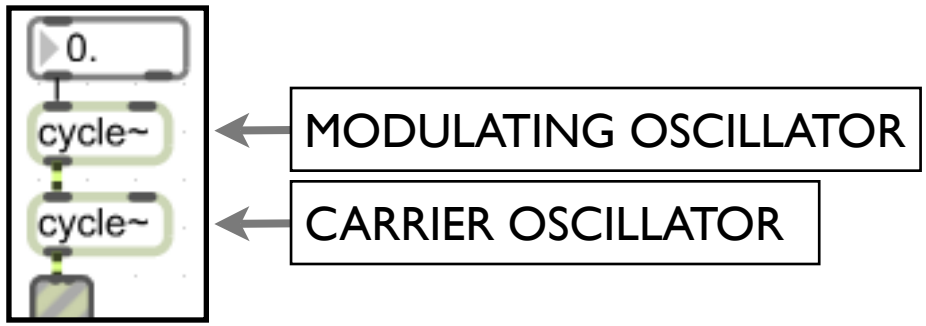
Ex.2

Now we'll look at Frequency Modulation. Starting from the same basic routine:



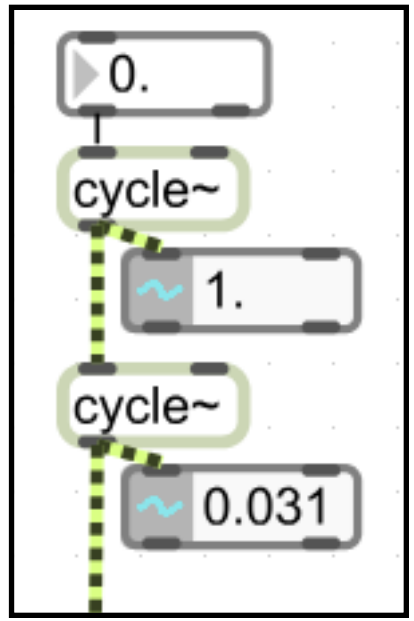
This time we need to change the pitch, which is of course controlled currently by the [float] object being sent to [cycle~]. Once again we could 'modulate' this frequency simply by changing the value in the [float] object, but we want it to change consistently. So we'll use an oscillator again.

1. Modify the routine as follows:



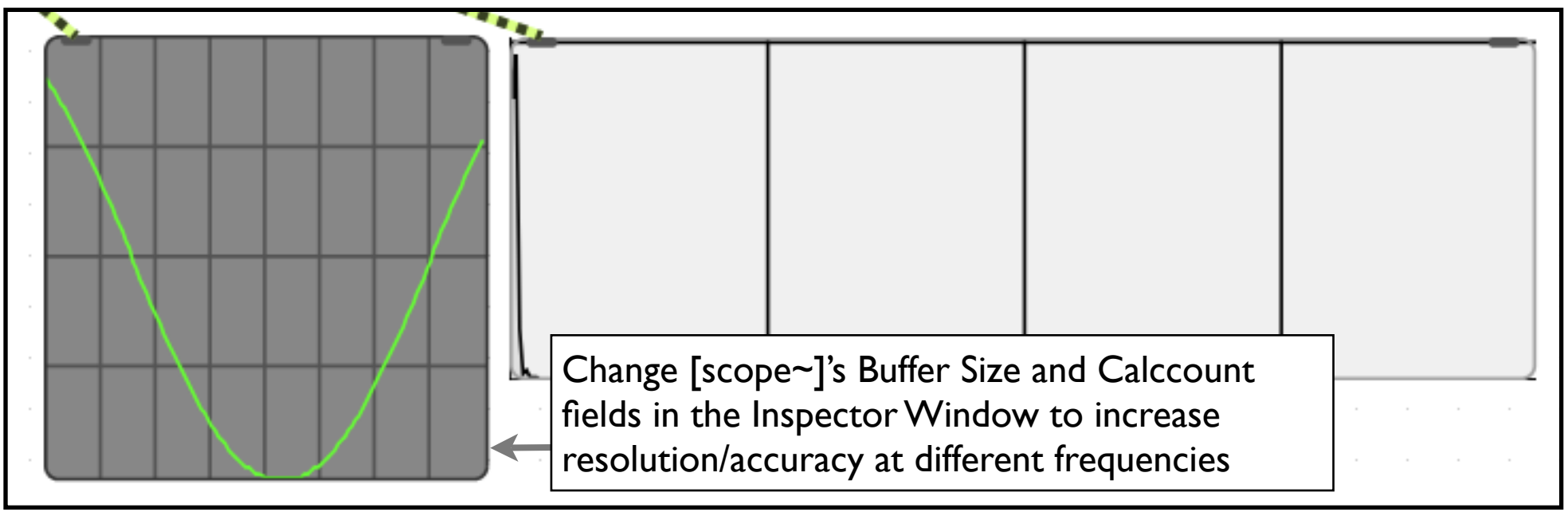
This would work in principle, but if you try it you won't hear anything because the carrier oscillator is being made to oscillate between -1Hz and 1Hz (i.e. inaudible frequencies). What we want is a starting frequency that the modulator can be made to oscillate around.

Interlude



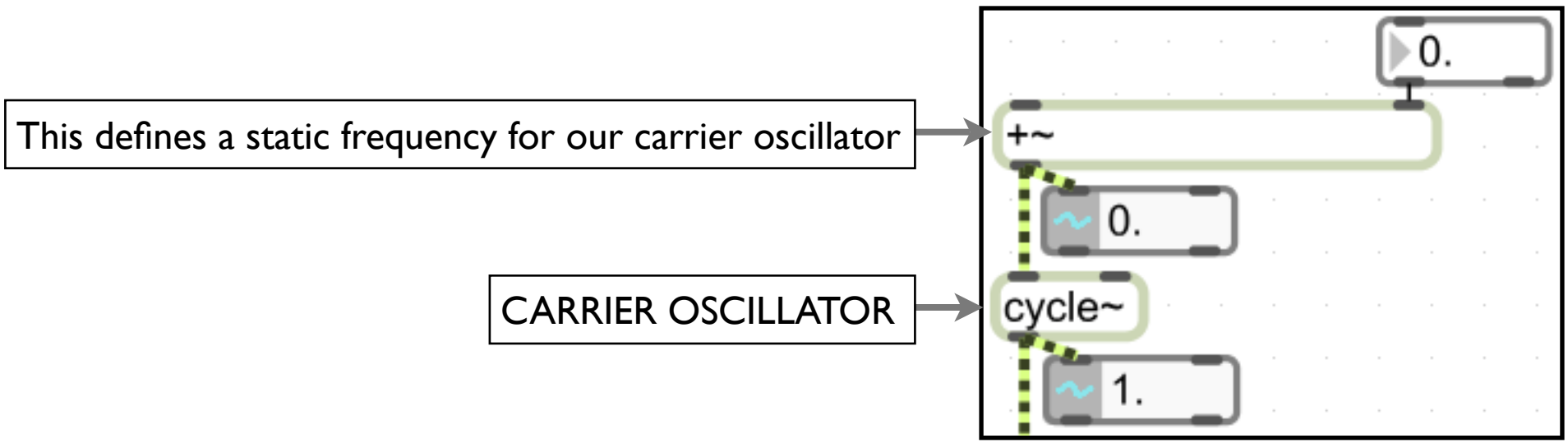
You can check to see what signal the two [cycle~] objects are outputting by using the [number~] object. This takes a sample of the signal at regular intervals. You can change how often it does this (i.e. make the reading more accurate) by changing the 'Update Interval in Milliseconds' field of its inspector window. 20 is a good value for here.

Other objects useful for signal display are [scope~] (an oscilloscope) and [spectroscope~] (spectrum analyser).



Ex.2 (cont)

2. For now, get rid of the top-most [cycle~] object (we'll put it back in a minute), modifying the routine as follows:

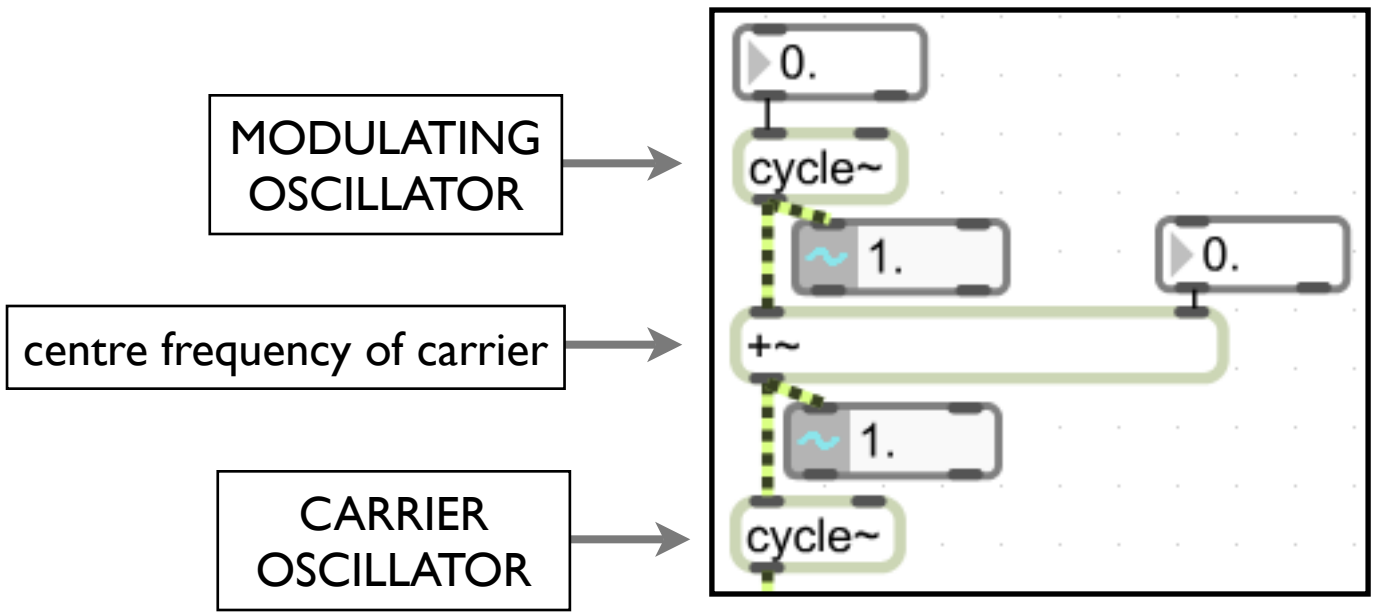


The above routine allows us to supply a fixed frequency to our [cycle~] object (i.e. we are adding nothing to our [float] object input.

Ex.2 (cont)

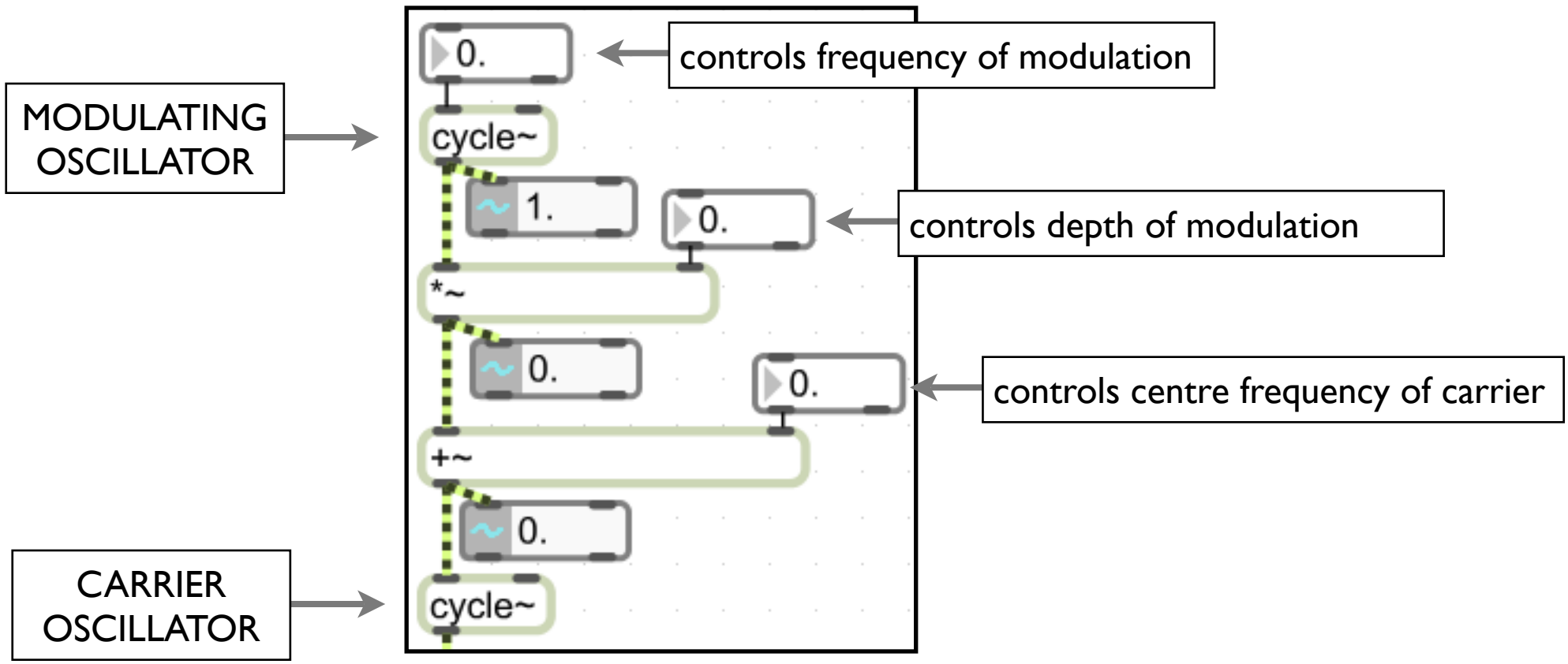
3. Now reintroduce the modulator oscillator [cycle~] object. The [cycle~] object outputs -1 to 1 which is added to our static frequency, allowing us to oscillate around a central frequency.

Keep the modulating [cycle~] at a low frequency for now so that you can clearly see (fia the [number~] object) and see what's happening.



Ex.2 (cont)

4. Now we've achieved vibrato or frequency modulation. The modulation isn't very deep though. You'll notice that we only move 1Hz above and below our centre frequency which isn't very dramatic. We can increase the depth by multiplying the modulating frequency:



Ex.2 (cont)

5. Explore the three controls for FM, noticing what happens as you change the frequency and depth of each oscillator.

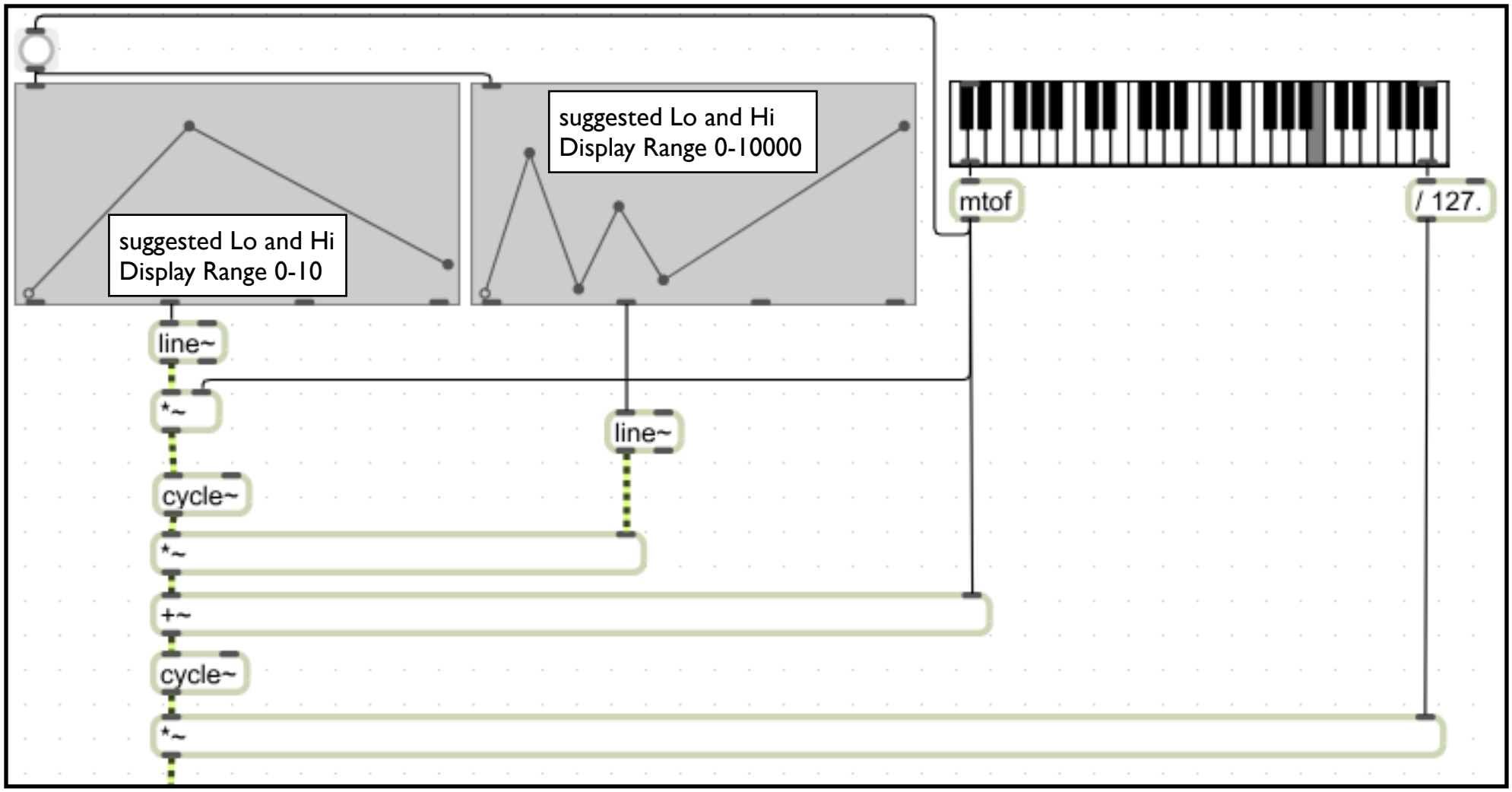
As with our explorations of AM synthesis, the timbre of the sound changes as the relationship between carrier and modulator changes.

6. Referring to our solutions for subtractive and amplitude modulation, attempt the following:

- Control the frequency of the carrier oscillator using a [kslider] object or MIDI keyboard
- Control the overall level of output using the right-hand outlet of the [kslider]
- ensure a consistent relationship between carrier and modulating frequencies;
- permit an evolving timbre by using [line] or [line~]/[function] objects, triggered by the [kslider], to control the carrier/modulation ratio and the modulation depth.

Ex.2 (cont)

Solution:



Ex.2 (cont)

You could of course use a [cycle~] object to modulate the depth of modulation in a more consistent way (though you'll need to work with its output to get an appropriate range of values).

More information about FM synthesis can be found on the MUST2004 site at this link: http://www.mti.dmu.ac.uk/must/MUST2004/supplemental/FreqMod.pdf. This discusses index of modulation and harmonicity which we won't discuss here.