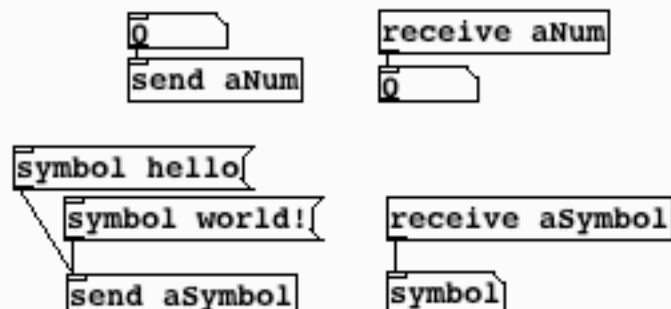
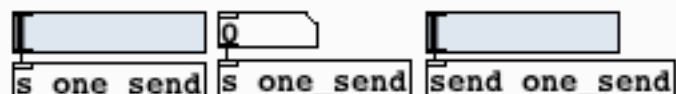


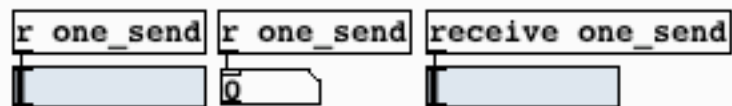
Send-N-Receive



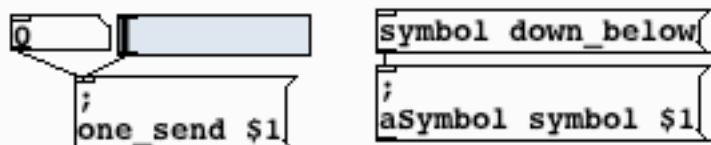
Multiple [send]s with the same name



received by [receive]s of the same name.



You can make message boxes into sends
(look above at previous examples):



Getting_your_message_across

The [send] and [receive] objects are useful for linking objects and subroutines together in a single canvas or throughout multiple patches. Understanding these objects will help 'clean up' your patches and allow you to communicate messages between various modules that you build or between objects where patch cords are not possible.

In the [send] and [receive] help files, you will see message sent within a single patch - from a [send] object to a [receive] object in the same patch. But the functionality of these objects goes far beyond the scope of a single patch.

Broadcasting_with_sends_and_receives

A [send] object is a little like a radio station...it broadcasts information throughout Pd, for all the [receives] to hear! A [receive] object is a little like your own AM/FM tuner...it will receive messages from all the [send] objects that it is 'tuned' to. Hence, naming your [send] and [receive] objects becomes very important. A [receive] object named "dave" (as in [receive dave]) will pick up all messages from [send dave], regardless of which patch it is in. In this manner, you can get messages across multiple patches, or into abstractions or subpatches -- without patch cords.