BASICS OF INSTRUMENTATION
LABVIEW EXERCISES

As a reference, download the *LabVIEW Fundamentals* manual from the National Instruments website
`http://www.ni.com/pdf/manuals/374029a.pdf`.
Additionally, the best guide is always the LabVIEW help. Try Ctrl+?

Parallel loops

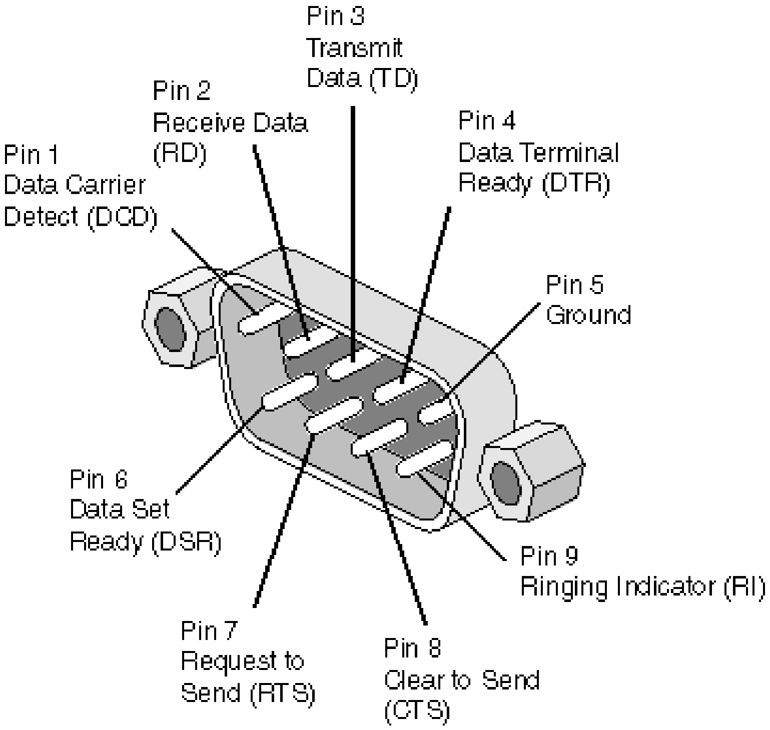
Queues. Make two parallel while loops. In one loop generate random double precision numbers in the range 0-1. At each iteration display the random number on the front panel and also add the element to a queue. In the second while loop read the last element in the queue. If the element is larger than 0.5, then display a green led on the front panel, otherwise make it red. Run both while loops once every 500ms.
Hint: look at the help for *Obtain queue*, *(Lossy) enqueue* and *Preview queue element*. Look at the example for the *Obtain queue* node. Ignore the way the example vi closes, it generates an error on purpose which is not nice... just press abort for now.

Stop parallel loops. How would you nicely interrupt the execution of parallel loops with only one stop button? There are several ways; first do it with queues. It works well but it looks unnecessarily complicated, right? So find and try at least two simpler solutions.
Hint: Google is your friend.

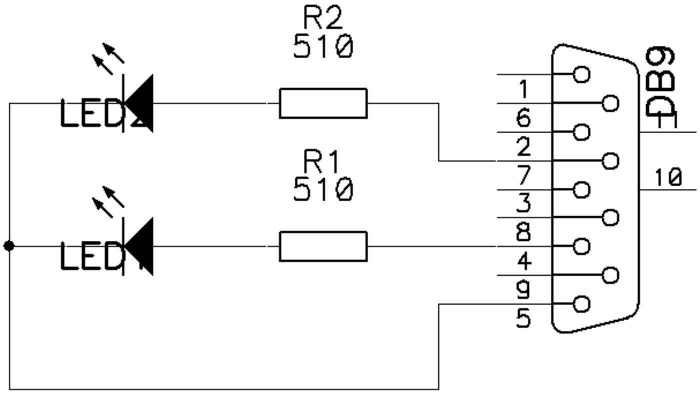
Data acquisition and control

Data acquisition: acquire some sound. Design a vi that records and saves a sound wave file you feed into the microphone port of your soundcard. Keep your vi as minimal as possible.
Hint: find the *Sound file write* node, open its detailed help, look at the example. Remove all the unnecessary stuff.

Event structures. First you'll need to connect two LED's to the serial port of your computer (see figure). Alessandro will show you the vi's you need to communicate to the serial port, then what you need to do is complete the vi in such a way that you can control the LED's from the front panel.
Create two boolean controls called RTS and DTR. In the block diagram add an event structure that does the following: in case of *value change* of RTS or DTR, feed into the RTS and DTR State property node the values of the corresponding controls (you'll need to convert boolean values into numeric 0,1). In case of timeout (a few seconds), do nothing. Run the vi and see what happens when you change the values of the RTS and DTR switches. Improve your vi by enclosing the event structure into a while loop.
Hint: read about event structures in the LabVIEW help or in LabVIEW Fundamentals. Try to right click on the event structure and select *Edit events handled by this case.* Make sure that the correct VISA resource name is selected (it should be COM1, or something similar).



(A) DB9 serial port pinout.



(B) Circuit to build in order to control two LED's through pins 4 and 7.