Homework #2: Laboratory Exercise 1
Due Date: April 20, 2006

Laboratory Exercise 1: Dynamic Range Control

Problem 1. [50 Points]

Completion of the following exercises entails making modifications to the compressor plugin 'Compressor', for which source code can be downloaded from the class website, `http://ccrma.stanford.edu/courses/424/spring-2005/handouts-2005.html`.
All necessary materials are contained in the file `lab1.tar.gz`.

Solutions for each excercise should include the source files `Compressor.cpp` and `Compressor.hpp`, suitably modified. The files must be able to be compiled. Additional write-up is required for some sections of this problem.

1(a). [10 Points] The stock compressor plugin uses a peak detector. Modify the code to use RMS detection. You will need to remove the controls for attack and release times, and add a control for the RMS integration time.

1(b). [10 Points] Edit the peak-detection code in the stock plugin so that the detector uses release-to-threshold, rather than release-to-zero. With the compression ratio set to infinity, the threshold of compression set at $-20dB$, and the input and output gains set to unity, write an expression for the compressor gain $\Phi_F(t), t > 0$, for the input signal $x(t) = 1 - u(t)$, using the stock detector. Repeat for the release-to-threshold detector.

1(c). [10 Points] Modify the stock compressor so that the release becomes program-dependent. Make the release behavior such that the release time is approximately one second following a sustained high input level, and make the release time following transients controllable using the existing slider for release time. Turn in plots showing the compressor's response to the signal `tdiff.wav`, with the fast release component set to 100mS.

1(d). [10 Points] Add a soft knee to the compressor's static compression function. Add a control so that the width of the soft knee can be set between zero and twenty dB.

1(e). [10 Points] Convert the compressor into a noise gate. Use the existing ratio control to control the plugin's *downwards expansion* ratio.

Problem 2. [**40 Points**]

For this problem, you will be asked to analyze the mystery plugin `DRC.dll`, which is contained in the file `lab1.tar.gz`. You may create any test signals you need to analyze the plugin, and may use MATLAB to study the output of the plugin.

2(a). [**4 Points**] Determine the threshold of compression and the compression ratio.

2(b). [**10 Points**] Write pseudo-code for the static gain functions Φ_F and Φ_B that would be needed to implement this compressor using feedforward or feedback topologies.

2(c). [**6 Points**] Determine whether the plugin uses RMS or peak detection.

2(d). [**6 Points**] Determine the approximate attack and release times for this plugin.

2(e). [**14 Points**] Write pseudo-code for the signal estimator that would be necessary to implement this plugin for the feedforward case (7 Points) and the feedback case (7 Points).

Hint: Unlike many audio-processing devices, the mystery plugin DRC *will support the entire spectral band from DC to the Nyquist limit.*