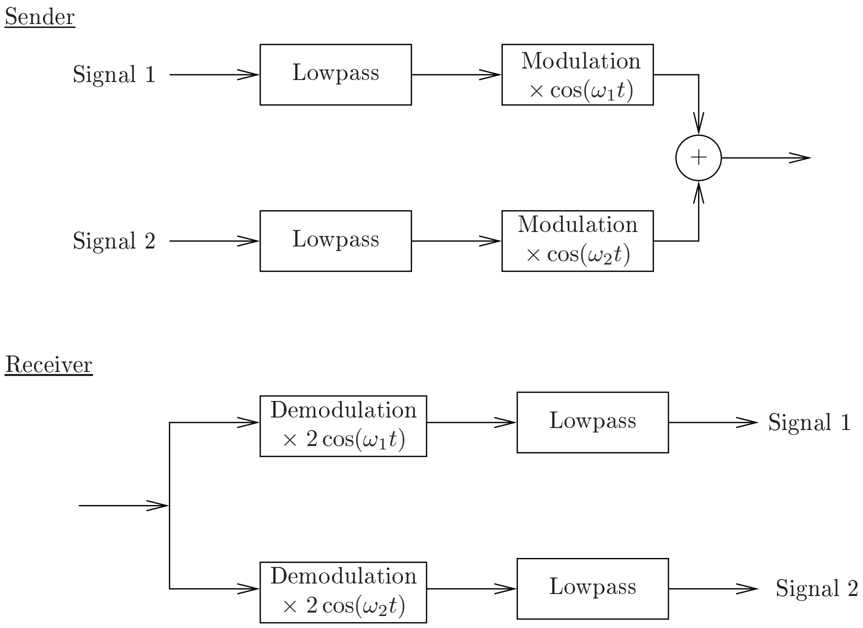
Project Digital Signal Processing

The objective of this project is the transmission of two signals over a common medium through modulation. At the senders side the signals are low pass filtered, modulated and added. The receiver does the demodulation and a second low pass filtering.



Assignment

- Generate two discrete signals, e.g. a sine, saw or rectangular wave or a music signal. If you convert the signals to an audio file you are able check the effect of the operations by listening.
- Apply a low pass filter with a suitable cutoff frequency to your signals (see exercise digital low pass filter). This way you can control the band limitation of the signals. Use discrete convolution in time domain first. If that works, try fast convolution with FFT. The results should be identical up to rounding errors.
- Optional: Increase the sampling rate such that the conditions of the sampling theorem still hold after modulation with a high frequency (see exercise changing sampling rate).
- Modulate both signals by multiplication with $\cos(\omega_1 t)$ and $\cos(\omega_2 t)$ where ω_1 and ω_2 are such that the results do not overlap in frequency domain, see modulation theorem.
- Add both signals.

- Reconstruct both signals with demodulation. In order to obtain the first signal, you have to multipliy the sum with $2\cos(\omega_1 t)$ and do low pass filtering. The second signal is obtained in the same way by multiplying with $2\cos(\omega_2 t)$.

- Compare the reconstructed and the original (low pass filtered) signals. A better low pass filter with more coefficients costs more computing time but leads to smaller differences between input and output signal. Do some experiments with different filter lengths, cut off frequencies and modulation frequencies. Make some tests what happens if the frequency bands overlap. It is interesting to hear the errors in the reconstructed music signals.

- Optional: Use quadrature amplitude modulation to transfer *two* signals in the same frequency band. All you have to do is to modulate/demodulate the first signal with $\cos(\hat{\omega}t)$ and the second with $\sin(\hat{\omega}t)$.

- Optional: Use single side band amplitude modulation to reduce bandwidth in signal transmission. The theoretical background of this method is described in the lecture notes. The actual implementation requires a Hilbert transform, which is nothing but a discrete convolution with a simple impulse response. As you have already an implementation of fast convolution, the additional effort is small.

Tipps for debugging

- Test each module separately:

  - Generation of signals and access to sound files
  - DFT
  - FFT
  - Computation of the coefficients of the low pass filter
  - Discrete convolution in time domain
  - Discrete convolution in frequency domain via DFT or FFT
  - Modulation / demodulation
  - Changing sampling frequency
  - Overall system

  Agree in your team on interfaces between the modules. That way you can divide up work and develop modules independently and in parallel.

- You can check "by hand" whether your DFT works if you use sinuoids or a mixture of two sine signals with different frequencies. Further you can apply DFT and IDFT successively and verify that the output is identical to the input up to rounding errors.

- If you use integer sample values and round the result to integers you don't have to deal with rounding errors.

- FFT and IFFT have to give identical results as DFT and IDFT up to rounding errors. Recursive and iterative FFT have to give identical results because exactly the same arithmetic operations are executed. Test this with (many) random signals.

- Make (very) sure that in your implementation of the FFT no sine and cosine terms are evaluated. They have to be precomputed in advance when the system starts up, otherwise FFT will be extremely slow.

- You can test your low pass filter with an (infinitely long) sine signal. If the frequency of the sine signal is above the cutoff frequency of the filter, the output has to be nearly zero. If it is below, the sine signal should pass almost unchanged. If more filter coefficients are used, more accurate results can be expected. Make sure that you can modify the number of filter coefficients in your program easily with one parameter. Using music signals you can test the result by listening.

- Compare your fast convolution with convolution in time domain. The results have to be identical up to rounding errors. Implement fast convolution first with DFT and replace DFT later on with FFT. This reduces computing time, the results are the same up to rounding errors.

- Test the overall system by transmitting only a single signal first. After modulation, demodulation and low pass filtering the output signal has to be close to the low pass filtered input signal. Deviations are caused only by the finite length low pass filter and a delay due to causality. Test this with sine waves and with music signals by listening.

- Draw the frequency bands of the modulated and filtered signals on the frequency axis. That way you can see easily if the bands overlap and if the conditions of the sampling theorem are satisfied.

- Try to have a running system fast and optimize only after it runs stable. Work first with synthetic, band limited signals like sinoids such that no low pass filtering at the senders side is necessary. Make sure not to violate the sampling theorem after modulation. Use convolution in time domain first and if that works, switch to fast convolution with DFT and FFT at the receiver. If that works, use low pass filtering at the sender too and do experiments with music sigals. If you still have time, try to change the sampling rate to have more space on the frequency axis.

Written report

- Please submit all your programs. Structure the code such that modules are well separated in different files, e.g. FFT, slow/fast convolution, low pass filter, modulation, etc.

- If not all parts work, please document where problems occur.

- Document your experiments and results. Try to explain your observations e.g. computing time, errors depending on filter length, errors with overlapping frequency bands, errors if sampling theorem is violated, etc. Please document also if your experimental results do not meet your expectations from theory.

- Explain the theoretical background very briefly in your own words.
- Length of the report should be around 8 pages.
- Do not forget to write your name and matriculation number on your report.
- Plase fill out the check list and submit it together with your report and programs.

Project presentation

- 20 minutes for your presentation and 5-10 minutes for questions.
- Very brief summary of theoretical background (two sentences).
- Focus on your own results: What was achieved, what was challenging?
- Show some experimental results (e.g. timings, different modulation frequencies with potentially overlapping frequency bands, reconstruction errors due to finite length filters or aliasing, effect of Hamming window, acoustic effects caused by filtering or Hilbert transform, etc.)
- Graphical presentation with charts instead of columns of numbers. Interpret your results, do they match your theoretical expectations?
- Show source code only upon request.
- Please attend the presentations of other teams at least in your own session and participate actively.

Conditions

- Teamwork is allowed with up to 3 persons in each team.