Signal and Data Analysis

Exercise 6

9.5.2008

Dr. T. Dietel, Prof. Dr. Ch. Weinheimer, Prof. Dr. J.P. Wessels

1 Nyquist Theorem

Write a program that calculates $x(t) = \sin \omega t$ for N evenly spaced time points between 0 s and 1 s. The program should print the time t and $x(t)$ of each sample on one line. For N=4 and $f = \frac{\omega}{2\pi} = 1\,\mathrm{Hz}$, the output should look like this:

```
0 0
0.25 1
0.5 1.22461e-16
0.75 -1
```

- Run the program for $N = 8$ and $N = 32$ with $f = 1.5\,\mathrm{Hz}, 3\,\mathrm{Hz}$ and $5\,\mathrm{Hz}$ and write the data to files.
- Visualize the sampled data using gnuplot.
- Run a Fast-Fourier-Transform on the data using the example program `fft`. The output file will contain a table of frequencies and the magnitude and phase of the complex FFT coefficients for these frequencies.
- Visualize the Fourier transform of the original sine wave.
- Run the inverse FFT to regain the original signal using `fft -i`.
- Visualize the reconstructed input signal.

2 Simulation of Analog Filter

The effect of an analog filter on a digitized signal can be simulated by transforming from the time into the frequency domain, applying the transfer function $G(s)$ of the filter and transforming back into the time domain.

(a) Use the program `exp` to generate a digitized signal $x(t) = 1 \cdot exp(t/0.1\,\text{s})$ with 1024 samples and a total duration of 1 s. Plot the signal shape.

(b) Fourier-transform the signal into the frequency domain using the program `fft`.

(c) Write a program `butterworth_lowpass.cc` that applies a 2nd order Butterworth low pass filter with a corner frequency of 20 and 200 Hz.

(d) Transform the spectrum back into the time domain using `fft -i` and display the filtered signal.

Note

The transfer function of a second order Butterworth filter is:

$G(s) = \frac{1}{1+\left(\frac{\omega}{\omega_c}\right)\sqrt{2}+\left(\frac{\omega}{\omega_c}\right)^2}$

3 Useful Commands

3.1 Example Programs

The example programs are available at

```
http://qqp.uni-muenster.de/~diete_00/signals08/sheet6-examples.tgz
```

After downloading the code to a Linux computer, the file can be unpacked with

```
tar -xvzf sheet6-examples.tgz
```

You will find the examples in the directory `sheet6`.

3.2 Account

You can log into the computer `lambda.uni-muenster.de` to compile and run the example programs given in this example. Use your ZIV username and password.

3.3 C++ Compiler

To compile C++ code in `name.cc` to produce an executable `name`:

```
g++ -O2 -lm -o name name.cc
```

The program can then be executed with `./name`.
You can use the example programs as a starting point for the exercises.

3.4 Input/Output Redirection

The example programs read from standard input and write to standard output.
To write the output of the program `exp` to a file, run:

```
./exp > data
```

To read the same data into the Fourier transform program and write the output to another file:

```
./fft < data > data.fft
```

Commands can also be chained together using pipes:

```
./exp | ./fft > data.fft
```

3.5 gnuplot

The program `gnuplot` can display data that is stored in files. To start the program, enter:

```
gnuplot
```

You will get a prompt where you can enter commands. To enter the online help, type `help`.
The most useful command for the beginning is to plot the second versus the first column from the file `datafile`:

```
plot "datafile"
```

Other columns can also be plotted. To draw e.g. the first versus the third column, use:

```
plot "datafile" (\$3):(\$1)
```

More information can be found at:

3.6 Problems

In case of problems, email me at `tom@dietel.net` or find me in room 218.