ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE
School of Computer and Communication Sciences

Software-Defined Radio:	Final Exam
A Hands-On Course	December 16, 2015

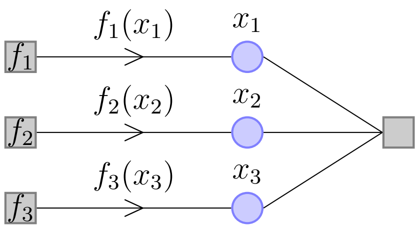
Name:

Note:

- You have 2 h 45 min to work at the exam.
- The exam is closed book, but your are allowed two pages of handwritten notes. You are only allowed to use the workstations in the laboratory (not your own laptops). Resources from the internet as well as code written outside this exam are not allowed.
- There is a `MATLAB` part and a paper-and-pencil part.
- The code will be evaluated according to the usual criteria, namely correctness, speed, form, and readability. Short comments that allow us to follow what you are doing will improve readability.
- Start by downloading from Moodle and unzipping the file `sdr_final_2015.tar.gz`. The files for Problem n, $n = 2, 3, 4$, are found in subfolder pn.
- You will upload to Moodle your solution to the problems that require writing `MATLAB` code. Do so in a **single archive**.
- The problems can be solved in any order.

Problem 1. (10p) *(LDPC-Decoder)* (Paper and Pencil)

Consider a ternary parity check code described by the factor graph below. The code is linear over GF(3), that is, the alphabet is $\{0, 1, 2\}$ and the operations are modulo 3.



$$f_4 = \begin{cases} 1 & \text{if } x_1 + x_2 + x_3 = 0 \text{ (modulo 3)} \\ 0 & \text{otherwise} \end{cases}$$

(a) Write a parity check matrix H for this code.

(b) Write a socket matrix S for this code.

(c) The function $f_1(x_1)$ is defined as follows:

$$f_1(x_1) = \begin{cases} 0 & \text{if } x_1 = 0 \\ 2 & \text{if } x_1 = 1 \\ 1 & \text{if } x_1 = 2 \ . \end{cases}$$

We abbreviate the definition above as $f_1(x_1) = (0, 2, 1)$. In a similar way, we define $f_2(x_2) = (1, 1, 0)$ and $f_3(x_3) = (0, 1, 0)$.

(i) Write down the expression for the message $F_{4\to3}(x_3)$.

(ii) Evaluate $F_{4\to3}(2)$.

Problem 2. (15p) *(Sampling)* (`MATLAB`)

The variable `samples` in the file `samples.mat` contains the samples of a signal. The sampling time is $T_s = 10^{-6}$ seconds and satisfies the conditions of the sampling theorem. Assume that the first sample is taken at time $t = 0$.

(a) Generate a vector of sampling times for 1024 equally spaced samples within the time interval where the signal is defined.

(b) Making use of the sampling theorem, compute these 1024 new samples of the signal. You should avoid loops.

(c) On the same figure, plot the new and the original samples. Use a line plot for the new samples, and circles for the original ones. Verify that the initial samples lie on the new signal's curve. The units of the time line should be in seconds.

Note: You might find useful one of the commands `meshgrid`, `ndgrid` or `repmat`.

Problem 3. (15p) *(Rotating constellation)* (`MATLAB`)

Let $s_1, s_2, \ldots, s_l$ be a preamble consisting of symbols from an 8-PSK constellation, and let $s_{l+1}, \ldots, s_{l+n}$ be a sequence of information symbols from the same constellation. By loading `data.mat` you obtain two vectors. The first is called `preamble`. It contains $s_1, s_2, \ldots, s_l$. The second, named `received`, is the received sequence that corresponds to the entire sequence $s_1, \ldots, s_l, s_{l+1}, \ldots, s_{l+n}$. It is the sampled output of the matched filter. We assume that the receiver front-end knows the carrier frequency used by the transmitter and that the sampling at the matched-filter output is done at the correct times.

(a) By plotting the elements of `received` in the complex plane (using the command `plot(received,'*')`) we expect to see a noisy 8-PSK constellation, but it is not the case. In fact there is a small rotation by an angle that increases linearly at every new symbol. Explain, in a sentence or two, what could cause this rotation. Write your explanation as a comment in the code (see next question).

(b) Write a `MATLAB` code that you name `pskReceiver.m`. With the aid of `preamble`, your code should remove the rotation and obtain the sequence of noisy 8-PSK symbols. Plot the resulting noisy symbols in the complex plane.

Problem 4. (20p) *(OFDM)* (`MATLAB`)

The file `ofdm_rx_signal.mat` contains the (noisy) received symbols of an OFDM signal transmitted over a discrete-time AWGN channel. However, the receiver is not yet synchronized, which means that the first received symbol does not correspond to the start of the cyclic prefix of an OFDM super-symbol. The OFDM transmitter uses a total number of 256 carriers and a cyclic prefix of length 25.

(a) In the received data, find the index of the beginning of the first complete cyclic prefix. Loops are allowed for solving this question. Hint 1: Use the known fact that there is a cyclic prefix in front of each OFDM super-symbol. Due to the noise, you cannot rely on a single OFDM super-symbol. Hint 2: To help debug and test your solution to this and the next question, we provide a "clean" received OFDM signal stored in `clean_rx_signal.mat` which is noiseless and starts with a cyclic prefix. You can cut this signal as you see fit to test your code.

(b) Once you have found the first cyclic prefix, remove all the cyclic prefixes, take the FFT of each complete super-symbol, and plot the resulting QAM symbols in the complex plane. (You should be able to recognize a noisy signal constellation.) Note that, at the end of the received data, some symbols could be missing as well.