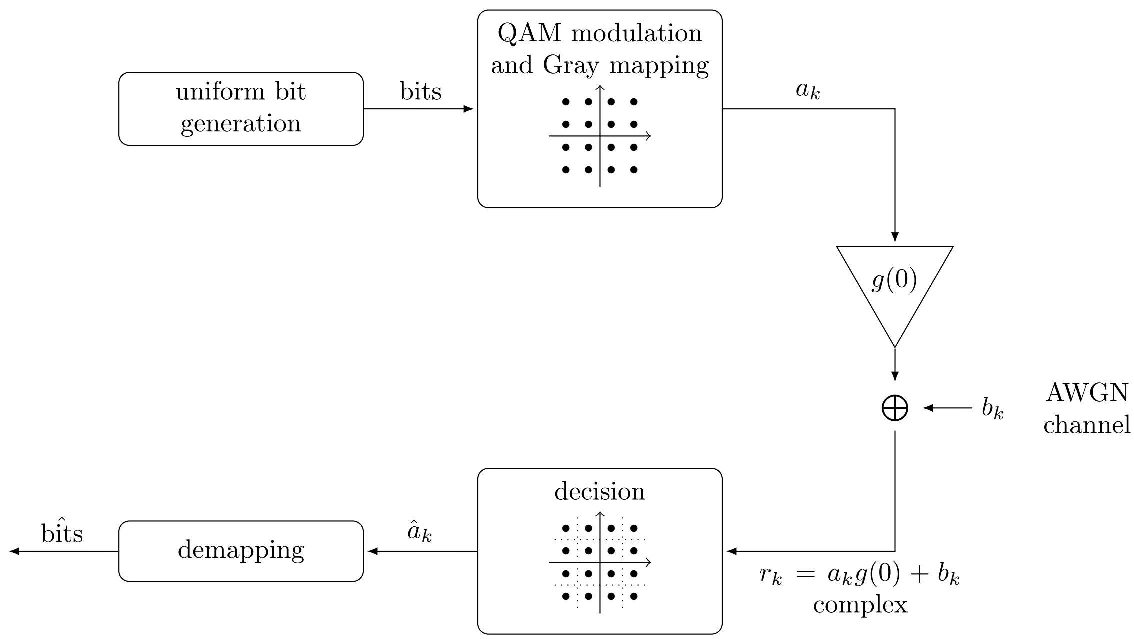
Digital Communications project
Part I: M-QAM Communication

The "Digital Communications" module consists of 6 lab sessions to be performed in MatLab. Attendance is mandatory and any unjustified absence will result in a systematic loss of points in the grade. If you cannot be present because of medical reasons, please contact your instructor rapidly.
During the lab, you will need to validate some ***checkpoints*** *with the instructor, which will affect your grade.*
At the end of each session and before leaving the lab room, you should send a progress report to the instructor (by e-mail or through Moodle) including MatLab files and a progress report. The final report of the TP must be sent on the last session.



The objective of this project is to test the performance of QAM modulation over an Additive White Gaussian Noise channel by numerical simulations with the Monte-Carlo method.

1 QAM modulation and Gray labelling

a) How many bits are encoded in an M-QAM symbol? Draw a picture of 4-QAM and 16-QAM constellations with Gray labelling.

b) Write a program that generates the square constellation M-QAM, stored in a matrix of size $\sqrt{M} \times \sqrt{M}$ which contains the **complex** values of the symbols $a_k = a_k^I + ja_k^Q$ with

$$a_k^I, a_k^Q \in \{\pm 1, \pm 3, \pm 5, \dots, \pm(\sqrt{M} - 1)\}.$$

This program must also generate a second matrix of size $\sqrt{M} \times \sqrt{M}$ which contains the corresponding binary labels of Gray's coding (stored in decimal).

Hint: You can use the MatLab functions `dec2bin`, `bin2dec`, `bi2de`, `de2bi`, `bitshift`, `bitxor`.

Hint: As an example, the two matrices of size 2×2 of the QPSK (or 4-QAM) are:

$$\text{mPoints} = \begin{pmatrix} -1+j & 1+j \\ -1-j & 1-j \end{pmatrix} \quad \text{et} \quad \text{mLabels} = \begin{pmatrix} 1 & 3 \\ 0 & 2 \end{pmatrix},$$

◇ **Checkpoint** 1 : Test the program for the cases $M \in \{4, 16, 64, 256\}$ using the MatLab code provided at the end of the document for the visualization of the constellation symbols and Gray labels.

2 Monte Carlo performance simulation

a) Write a program that simulates the M-QAM communication chain shown in the figure above (assuming an ideal Nyquist channel) for $M \in \{4, 16, 64, 256\}$.

◇ **Checkpoint** 2 : Check that the decision block returns the closest point in the constellation to the received point, and that the demodulation block returns the correct Gray label.

b) We will now evaluate the symbol error probability P_s and the bit error probability P_b numerically through a Monte Carlo simulation.
Supposing that the error probability we aim to determine is between 10^{-5} and 1, what is the expected number of samples required for the Monte Carlo simulation?

◇ **Checkpoint** 3 : Recall the expression for the average energy of an M-QAM constellation as a function of the size M.
Write the expression for the noise variance (per real dimension) as a function of the SNR E_b/N_0 (in dB).

c) Estimate P_b and P_s numerically for each value of the SNR $\left[\frac{E_b}{N_0}\right]_{\text{dB}} \in \{0, 1, 2, \ldots, 13\}$ and plot them (in log scale) as a function of $\left[\frac{E_b}{N_0}\right]_{\text{dB}}$.

◇ **Checkpoint** 4 : Compare the estimated curves with the theoretical curves. You can use the formula for the binary error probability $P_{b(min)}(e)$ parametrized by M, the constellation size (seen in class).

Hint: You can use the MatLab functions `qfunc`, `semilogy`.

Hint: When using Gray labelling, what is a (good approximation) for the relation between P_b and P_s?

◇ **Checkpoint** 5 : Compare M-QAM modulation for $M \in \{4, 16, 64, 256\}$ in terms of bit rate, bit error probability and spectral efficiency.

3 Complementary exercises

a) Replace the Gray mapping with an arbitrary mapping and check the impact of the mapping change on the binary error probability.

b) Add a phase error in addition to the complex noise, which simulates the synchronization errors: $g(0) = \exp(j\phi)$ and measure the degradation in terms of error probabilities for different values of ϕ.

```
% Visualize the M-QAM constellation and Gray labelling
%---------------------------------------------------------------
% M : constellation size
% mComplex : matrix containing the complex symbols
% of size (sqrt(M), sqrt(M))
% mGray : matrix containing Gray labels (in decimal)
% of size (sqrt(M), sqrt(M))

x = real(mPoints(:));
y = imag(mPoints(:));
z = mLabels(:);

% new figure
figure

% draw the points of the constellation
scatter(x,y,50,'b*');
axis([-sqrt(M) sqrt(M) -sqrt(M) sqrt(M)]);

% add Gray labelling
for k = 1 : M
    text(x(k)-0.6,y(k)+0.3,...
        dec2base(z(k),2,log2(M)),'Color',[1 0 0]);

end

% figure parameters
title('Gray Coding for M-QAM');
xlabel('I');
ylabel('Q');
grid on
```