

**INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA**  
**LICENCIATURA EM ENGENHARIA INFORMÁTICA E DE COMPUTADORES**  
**COMUNICAÇÃO DIGITAL**

**Verão 2021/2022 - Trabalho Prático (Módulo 2)**

**Data de publicação: 24 de maio de 2022**

**Data de entrega: 27 de junho de 2022**

---

**Objetivos:**

- Desenvolvimento de programas e aplicações em linguagem ‘C’ e ‘Python’.
  - Estudo e aplicação de conceitos fundamentais sobre SCD, cifra, codificação de canal e outras técnicas.
  - Estudo e aplicação de conceitos e desempenho de sistemas de transmissão digital, ao nível físico com código de linha e modulação digital.
  - Realização de comunicação entre dispositivos.
- 

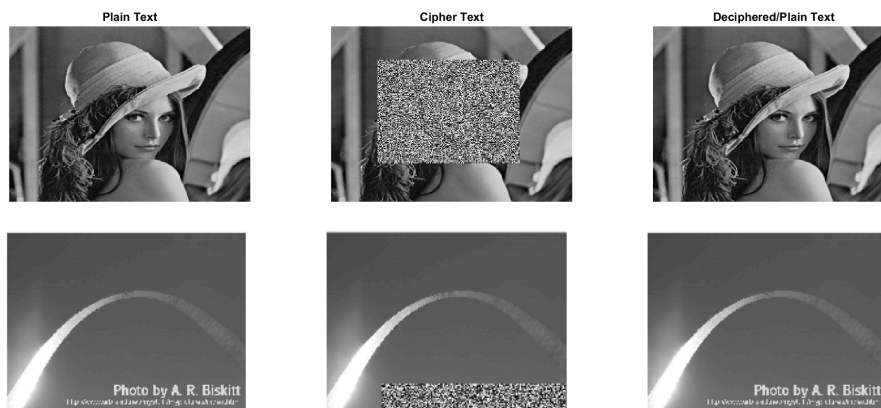
*O código desenvolvido e o respetivo relatório deverão ser entregues em formato eletrónico no sistema Moodle.  
A apresentação da resolução dos vários exercícios decorrerá em aula a definir em cada turma.*

1. **(C ou Python)** Considere a cifra de César ([https://pt.wikipedia.org/wiki/Cifra\\_de\\_César](https://pt.wikipedia.org/wiki/Cifra_de_César)) e a cifra de Vernam (*One Time Pad*), ([https://pt.wikipedia.org/wiki/One-time\\_pad](https://pt.wikipedia.org/wiki/One-time_pad)).

- Implemente as funções `ceasar_cipher`, `ceaser_decipher`, `vernam_cipher` e `vernam_decipher`, de forma que estas efetuem as suas ações, de forma genérica, sobre ficheiros de entrada, produzindo os respetivos ficheiros de saída.
- Para dois ficheiros à sua escolha do conjunto `CD_TestFiles.zip`, aplique ambas as cifras. Apresente:
  - o ficheiro original (texto em claro), o respetivo histograma e o valor da entropia;
  - o ficheiro cifrado (texto cifrado), o respetivo histograma e o valor da entropia;
  - o ficheiro decifrado (igual ao original), o respetivo histograma e o valor da entropia;
  - os valores do equívoco da chave e do equívoco da mensagem.

Comente os resultados obtidos. Nalguma situação indicada temos segurança perfeita?

- Implemente um cifrador/decifrador de imagens monocromáticas (em níveis de cinzento). A imagem de entrada (*plain text*) deverá ser cifrada na sua totalidade ou numa área retangular, definida pelo utilizador. A figura abaixo apresenta um exemplo desta funcionalidade. Apresente resultados experimentais que comprovem o funcionamento do conjunto cifrador/decifrador.



2. **(C ou Python)** Considere a implementação de códigos de controlo de erros com a técnica *Cyclic Redundancy Check* (CRC).

- Implemente as funções `crc_file_compute` e `crc_file_check`, as quais calculam e verificam os bits de paridade através de CRC, com polinómio gerador configurável. A função `crc_file_compute` recebe um ficheiro de entrada e produz o respetivo ficheiro de saída. A função `crc_file_check` efetua a verificação da integridade do ficheiro, através do cálculo do respetivo síndrome (detecção de erros).
- Apresente resultados experimentais que comprovem o funcionamento das funções desenvolvidas na alínea anterior, para as seguintes situações: ausência de bits em erro; percentagem de bits em erro de 0,01 %, 0,1 %, 0,5 %, 1 % e 5 %.
- Considere o par codificador/descodificador de código unário (*comma code*) desenvolvido no exercício 3 do primeiro módulo do trabalho prático. Integre este módulo de CRC no par codificador/descodificador. Apresente resultados que comprovem a correta aplicação da funcionalidade de deteção de erros com CRC.

3. **(Python)** Considere o diagrama de blocos no qual se representam os elementos da camada física de um Sistema de Comunicação Digital (SCD), com tempo de bit  $T_b = 1$  ms.

### Sistema de Comunicação Digital



O SCD tem as seguintes características:

- **Emissor** - O sinal  $x(t)$  presente na saída do emissor pode ser:  
NRZ-Unipolar -  $x_{NRZ}(t) = 5 \prod \left( \frac{t}{T_b} \right)$ .  
PSK -  $x_{PSK0}(t) = 2 \cos(2\pi 2000t)$  e  $x_{PSK1}(t) = -2 \cos(2\pi 2000t)$ .  
Na geração dos sinais, cada tempo de bit é representado por 10 amostras.
- **Canal de Comunicação** - O sinal de saída é  $y(t) = \alpha x(t) + n(t)$ , em que  $\alpha$  é um valor no intervalo  $]0, 1]$  e  $n(t)$  é o sinal de ruído. A relação sinal-ruído deverá ser parametrizável.
- **Recetor** - Este bloco é baseado no conceito da deteção coerente.

(a) Considere as especificações indicadas acima para o emissor e o recetor.

(i) Implemente as funções `NRZ_Coder` e `PSK_Modulator`, as quais recebem como parâmetros de entrada a sequência binária a transmitir e outras variáveis necessárias para o correto funcionamento do SCD. O parâmetro de retorno corresponde ao vetor com as amostras de  $x(t)$  correspondentes ao tempo de bit. Realize os respetivos recetores (as funções `NRZ_Decoder` e `PSK_Demodulator`). Explique sucintamente as opções tomadas.

(ii) Nas condições de canal ideal, apresente os sinais  $x(t)$ ,  $y(t)$  e a sequência binária recebida quando na entrada está a sequência binária 10110001.

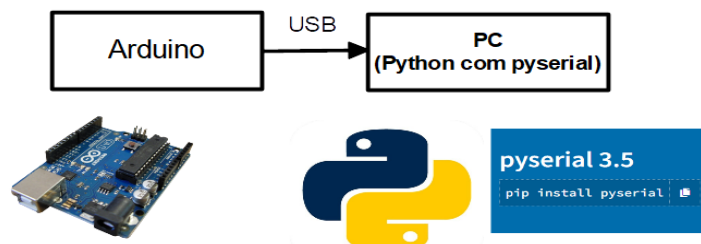
(b) Implemente a transmissão de conteúdos de ficheiros através do SCD desenvolvido na alínea (a) e cálculo do respetivo BER. Recorra aos ficheiros de `CD_TestFiles.zip`, para realizar a transmissão de conteúdos nas seguintes condições:

(i) Com  $a = 1$ , gere sinais de ruído  $n(t)$  com diferentes valores de energia. Apresente os valores de BER. Comente os resultados obtidos.

(ii) Com  $a < 1$  e com energia de ruído  $n(t)$  constante, apresente os valores de BER. Comente os resultados obtidos.

### 4. (Plataforma Arduino e Linguagem de Programação Python)

Recorrendo à plataforma Arduino, <https://pt.wikipedia.org/wiki/Arduino> e ao PC, usando o módulo `pyserial` da linguagem Python, disponível na página <https://pypi.org/project/pyserial>, estabeleça uma ligação via USB em modo *simplex*. O Arduino será emissor e o PC será recetor da informação, cujo conteúdo fica à sua escolha. A informação recebida será escrita na consola ou em ficheiro.



(a) Descreva o SCD realizado e o tipo de informação transmitida.

(b) Apresente resultados experimentais que comprovem o correto funcionamento do SCD. Em alternativa, o grupo de trabalho poderá demonstrar o funcionamento do SCD, em aula ou numa altura a agendar com o docente.

Para a elaboração deste módulo do trabalho prático, considere, entre outros, os recursos indicados em seguida.

Exercício 2:

- (1) <https://pypi.org/project/crc>
- (2) <https://barrgroup.com/downloads/code-crc-c>

Exercício 3:

- (1) <https://matplotlib.org>
- (2) <https://realpython.com/python-matplotlib-guide>

Exercício 4:

- (1) <https://www.arduino.cc>
- (2) <https://www.arduino.cc/en/software>

Seguem-se alguns aspetos relevantes sobre o relatório pretendido para este módulo do trabalho prático:

---

- (1) O relatório deve ser sucinto e organizado em quatro secções, uma por cada exercício do módulo do trabalho prático.
  - (2) Cada secção deve estar organizada em sub-secções, uma por cada alínea de cada exercício.
  - (3) Para cada alínea, o relatório deve ter resposta clara às questões colocadas no enunciado. Também deve conter todos os resultados experimentais solicitados e os respetivos comentários, análise e explicações consideradas essenciais.
  - (4) Nos exercícios em que se solicita a escrita de uma função ou o desenvolvimento de uma aplicação, o relatório deverá apresentar os principais critérios e escolhas efetuadas nesse desenvolvimento.
  - (5) O relatório não deve conter o código. Este deverá ser entregue em formato eletrónico em separado, devidamente comentado e organizado.
-