



# MESTRADO INTEGRADO EM TELECOMUNICAÇÃO E INFORMÁTICA

MICROCONTROLADORES

## **Trabalho prático 7**

**Controlo de acessos**

**Paulo Cardoso**

## Objetivo

Pretende-se alterar/acrescentar funcionalidades ao trabalho anterior.

## Descrição

Neste trabalho pretende-se:

- a) Permitir a introdução remota do código, via porta série. Depois de entrar num modo, local ou remoto, não é possível comutar até ser inserido o código completamente.
- b) Permitir a alteração do código apenas quando a porta está aberta. Implica que o *timer* do trinco seja por interrupção (porquê?)
- c) Implementação de um *timeout* para a inserção do código, local ou remotamente. Implica um *timer* a funcionar por interrupção (porquê?)
- d) Tal como já é possível para o código local, possibilidade de abortar a inserção do código remoto

## Metodologia

A implementação deste trabalho implica alterações ao código já existente, nomeadamente:

- i) Criar mais do que uma rotina para gerir os tempos de espera.
- ii) Implementar uma rotina **readSerial()** que recebe como parâmetro um *array* onde será colocada a password enviada pela porta sérieda

## Preparação

Cada grupo deverá entregar no início da próxima aula os novos fluxogramas do trabalho.

## Resultados

No fim do trabalho deverão enviar por *email* o código do trabalho, devidamente comentado.

## Duração do trabalho

Início: 17/12. Duração: Até ao fim do semestre (três aulas)

**Nota**

Em termos de pseudo-código, o algoritmo geral poderá ter a organização abaixo

PARA (sempre) FAZER

[Inicializações]

res = **detectKeys()**

SE (res != 0) ENTÃO // Uma tecla premida

res = **readPass()**

SE (res != 0) ENTÃO // Código de 4 dígitos bem inserido

res = **checkCode()**

SE (res != 0) // Password OK

[Abrir porta]

erros = 0

SENÃO

erros ++

SE (erros = 3)

**wait()**

SENÃO // Duas teclas premidas: mudança de password

**newPass()**

A rotina **readChar()** poderá ser assim:

**readChar()** //Esta rotina bloqueia até que seja inserido o ENTER

tecla = 0

res = 0

ENQUANTO (res != 1)

res = **detectKeys()**

SE (res = 0) //Operação abortada (dupla tecla premida)

tecla = 11

res = 1

SENAO //Operação normal (tecla incremento premida)

tecla++

SE (tecla = 10) ENTÃO tecla = 0

**display()**

**wait()**

```
wait()
```

RETORNA tecla

A rotina **detectKeys()** poderá ser assim:

```
detectKeys() //Esta rotina bloqueia até ser premida uma tecla (ou ambas)
```

```
tecla = 0
```

```
tecladupla = 0
```

```
ENQUANTO ( tecladupla = 0)
```

```
    SE ([tecla ENTER premida]) ENTÃO
```

```
        tecla=1
```

```
        tecladupla++
```

```
    SE ([tecla normal premida]) ENTÃO
```

```
        tecla=2
```

```
        tecladupla++
```

```
SE (tecladupla = 2)
```

```
    ENTÃO
```

```
        tecla = 0
```

```
RETORNA tecla
```