



MESTRADO INTEGRADO EM TELECOMUNICAÇÃO E INFORMÁTICA

MICROCONTROLADORES

Trabalho prático 6

Controlo de acessos

Paulo Cardoso

Objetivo

Pretende-se alterar/acrescentar funcionalidades ao trabalho anterior.

Descrição

Neste trabalho pretende-se:

- a) A alteração da rotina **wait()** de forma contar tempo com base num *timer*;
- b) Simular a existência de um trinco elétrico, num do pinos de GPIO;
- c) Bloquear o sistema transitoriamente caso existam três tentativas falhadas seguidas. No *display* deverá ser apresentado um traço horizontal;
- d) Possibilitar a alteração da *password* por defeito do sistema. Para isso, com o sistema em repouso, deverão ser premidos os dois botões simultaneamente. Caso estes dois botões sejam premidos a meio da inserção de um código, abortam a operação em curso.

Metodologia

Para a implementação deste trabalho, deverão ser realizadas alterações ao trabalho existente, nomeadamente

- i) Criação da função **detectKeys()** que deteta se ambas ou apenas uma tecla foi premida, retornando **0** se foram ambas premidas, **1** se foi premido o ENTER e **2** se foi premida a outra tecla;
- ii) **readChar()** deverá ser alterada para acomodar a função **detectKeys()**; A rotina retorna o número digitado, ou **11** caso ambas as teclas tenham sido premidas
- iii) Deverá também existir uma rotina **newPass()**, para definir uma nova *password*. Esta rotina recebe um apontador para a *array* da *password* atual, nada retornando;
- iv) Para evitar a duplicação de código *assembly* na introdução da *password*, este código atualmente na *main* deverá ir para uma função **readPass()**, que recebe como parâmetro um *array* para onde irá a *password* digitada e retorna **1** se a operação foi bem sucedida ou **0** se a operação foi abortada.

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

Duas a três aulas

NOTA: A título meramente indicativo, de seguida são apresentados alguns algoritmos



Algoritmos

O algoritmo geral poderá ter a organização indicada abaixo.

```

main()
    [Inicializações]
    PARA (sempre) FAZER
        res = detectKeys()
        SE (res != 0 )                // Uma tecla premida
            res = readPass()
            SE (res != 0)              // Código de 4 dígitos bem inserido
                res = checkCode()
                SE (res != 0 )        // Password OK
                    [Abrir porta]
                    erros = 0
                SENÃO                 // Password errada!
                    erros ++
                    SE (erros = 3) //Demasiadas tentativas erradas
                        wait()
            SENÃO                     // Duas teclas premidas: mudança de password
                newPass()

```

Uma outra implementação para o algoritmo geral é indicado abaixo. Neste caso a rotina **readPass()** tem de indicar também (no retorno) a passagem ao modo “alterar *password*”.

```

main()
    [Inicializações]
    PARA (sempre) FAZER
        res = readPass()
        CASO
            res=1                    // Código de 4 dígitos bem inserido
                res = checkCode()
                SE (res != 0 )        // Password OK
                    [Abrir porta]
                    erros = 0
                SENÃO                 // Password errada!
                    erros ++
                    SE (erros = 3) //Demasiadas tentativas erradas
                        wait()
            res=2                    // Duas teclas premidas: mudança de password
                newPass()

```

A rotina **detectKeys()** poderá ter o algoritmo apresentada abaixo.

```

detectKeys()           //Esta rotina bloqueia até ser premida uma tecla (ou ambas)
    tecla = 0
    tecladupla = 0
    ENQUANTO ( tecladupla = 0)
        SE ([tecla ENTER premida])
            tecla=1
            tecladupla++
        SE ([tecla normal premida])
            tecla=2
            tecladupla++
    SE (tecladupla = 2) tecla = 0
    RETORNA tecla

```

A rotina **readChar ()** poderá ter o algoritmo apresentada abaixo.

```
readChar ()           //Esta rotina bloqueia até que seja inserido o ENTER
    tecla = 0
    res = 0
    ENQUANTO (res != 1)           //Enquanto não for inserido o ENTER
        res = detectKeys ()
        CASO
            res = 0           //Operação abortada (dupla tecla premida)
                tecla = 11
                res = 1
            res = 2           //Operação normal (tecla incremento premida)
                tecla++
                SE (tecla = 10)
                    tecla = 0
                display ()
                wait ()
    wait ()
    RETORNA tecla
```
