

## U.C. Sistemas & Automação

### Trabalho Laboratorial

### Projeto do sistema de controlo de Misturador Industrial utilizando programação C *simpleCsim*

Armando Jorge Sousa

[asousa@fe.up.pt](mailto:asousa@fe.up.pt)

Paulo Costa

[paco@fe.up.pt](mailto:paco@fe.up.pt)

(versão 25-04-2018)

---

# 1. Objetivos e Introdução

Este trabalho pretende aprofundar a capacidade de codificar um controlador modelizado em Grafcet (incluindo funcionamento normal, supervisão e controlo) desta vez utilizando linguagem C.

O Caderno de Encargos (C.E.) é uma alteração do trabalho da misturadora.

Este trabalho laboratorial durará duas semanas (exceto “feriados”).

O programa “RemotePanel” (Figura 1) serve de sinótico e quadro de botões. A lógica de comando ficará no “SimpleCSimulator” que será desenvolvido pelos estudantes de acordo com a arquitetura apresentada na Figura 3 e as ligações mostradas na Figura 2.

O software utilizado é para Windows, livre e sem qualquer instalação.

No final deste trabalho laboratorial deve entregar a funcionar a misturadora respeitando as regras de produção sistemática de código do Grafcet. **Não basta funcionar, deve respeitar as normas de produção de código Grafcet.**

Relembre os sinais da misturadora na Figura 4 mas admita que os sinais “Bomba” e “V5” estão agora no mesmo sinal elétrico, tal como na Fig. 2.

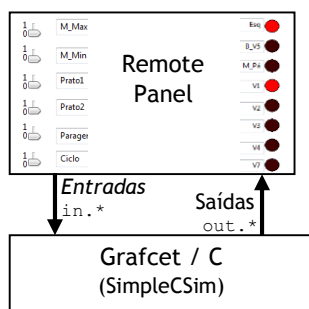


Figura 3 - Arquitetura

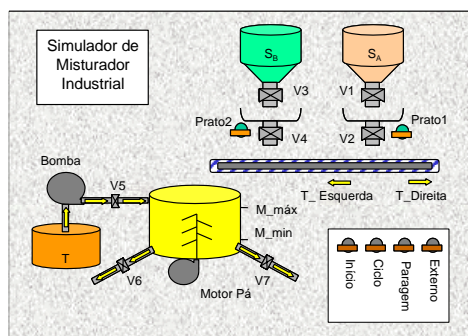


Figura 4 - Sinótico do misturador  
(não utilizado neste trabalho laboratorial)

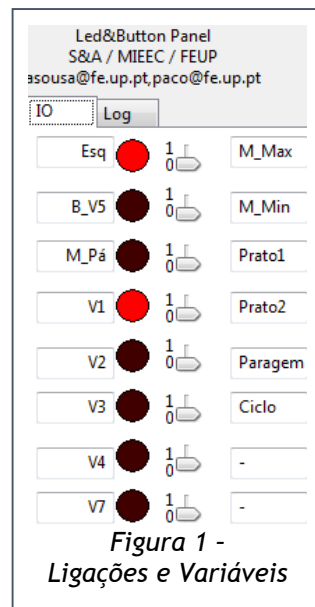


Figura 1 -  
Ligações e Variáveis

ESQUERDA out.b0  
BOMBA\_V5 out.b1  
MOTOR\_PA out.b2  
V1 out.b3  
V2 out.b4  
V3 out.b5  
V4 out.b6  
V7 out.b7

M\_MAX in.b0  
M\_MIN in.b1  
PRATO1 in.b2  
PRATO2 in.b3  
PARAGEM in.b4  
CICLO in.b5

Figura 2 -  
Ligações vistas do lado  
do SimpleCSim

## 2. Preparação, Procedimentos e Trabalho

Para a aula 1, estude todo o guião cuidadosamente e prepare o Grafcet de teste. Estude ainda as estruturas de dados a utilizar no programa em C (uma etapa, uma transição que dados utilizam, como se implementa uma ação, etc.). Planeie ainda como fazer debug. Planeie demonstrar ao professor que o Grafcet funciona.

Para a aula 2 prepare a solução do C.E. (utilize a norma do Grafcet).

## 3. Procedimentos para Trabalho

1. Descarregue do Moodle para uma diretoria nova (só sua, no drive F:) o \*.ZIP relativo ao Lab9
2. Descomprima o pacote criando as diretorias simpleCsim e DevCpp\*Portable.
3. Comece por testar a comunicação: dentro da diretoria simpleCsim, execute o RemotePanel.EXE e o simpleCsim 32b\_original.EXE que devem comunicar. Se necessário, dar permissões de comunicação ao nível do S.O. (tal como na Fig. 5; a imagem pode diferir conforme a versão do S.O.).

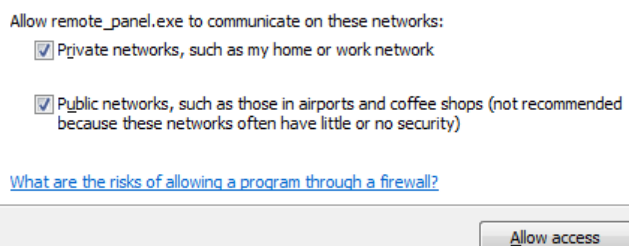
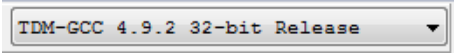


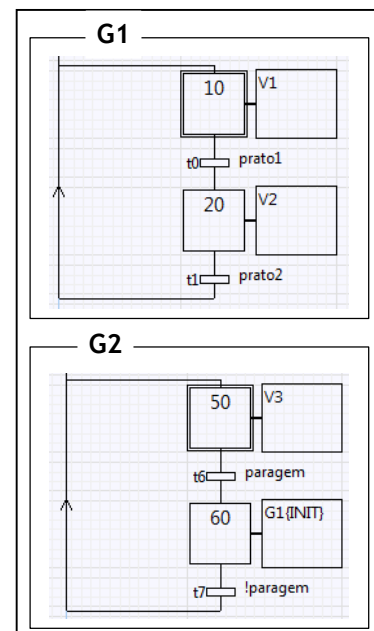
Figura 5 - Premir “Allow Access” na Firewall

4. Abra o ambiente integrado de desenvolvimento (IDE) executando o `devcppPortable.exe`
  5. Dentro do IDE DevCpp, escolha File -> Open; escreva \*.dev e depois escolha o `simpleCsim`
  6. Se estiver em Windows XP ou outro sistema operativo de 32 bits configure para  (em cima, perto da direita do ecrã)
  7. Escolha Execute -> Rebuild\_All
  8. Prima F11 sempre que quiser compilar e executar; prima ESC para sair do executável e regressar ao IDE
  9. Prima F5 para configurar o IDE para *debugging* (aceite a opção mostrada) e sempre que quiser utilizar o debugger; teste o debugger por exemplo colocando um *breakpoint* por exemplo na linha 80; inspecione a variável ESQUERDA antes e depois da atribuição; explore o *debugger*
- Depois dos passos acima mostrados, deve ser capaz de desenvolver o código da aula.

**Atenção:** Alterar sempre e só o ficheiro `main.c`

### Semana 1:

- Crie as estruturas de dados que entender necessárias; por simplicidade não utilize alocação dinâmica de memória, utilize sim, vetores estáticos sobredimensionados (exemplo: 128 estados e 128 transições); tenha atenção que TODAS as variáveis locais devem obrigatoriamente ser inicializadas manualmente
- Não esqueça de retirar ou comentar todo o código de teste
- Implemente o Grafcet G1 de teste mostrado na figura ao lado, com a notação da norma IEC60848 mas com transições em linguagem C
- Implemente um sistema de debug do Grafcet que imprima: a situação de cada um dos Grafcets (isto é, imprimir os números das etapas ativas) e as entradas e saídas ativas neste instante
- Implemente agora o Grafcet G2 de teste que inclui comando sobre G1
- Identifique com um comentário “///// Final do cálculo das transições disparadas /////”
- Demonstre ao professor as funcionalidades atingidas antes do final da aula
- Não saia da sala sem **responder às questões do professor**
- Até à data mostrada no Moodle, submeta os elementos abaixo pedidos (secção 5)
- Sal guarde o seu trabalho
- Apague a sua diretoria



### Semana 2:

- Desenvolva o sistema para cumprir com todos os requisitos do caderno de encargos ao nível do funcionamento normal (secção 4, itens de 1 a 8)
- Complete com a especificação adicional de controlo supervisorio (secção 4, item 9)
- Teste todo o sistema
- Demonstre ao professor as funcionalidades atingidas antes do final da aula; **responda às questões do professor**
- Até à data mostrada no Moodle, submeta os elementos abaixo pedidos (secção 5)
- Sal guarde o seu trabalho
- Apague a sua diretoria

## 4. Caderno encargos do misturador industrial alterado

---

Após arranque do sistema, esperar o ciclo de funcionamento normal do misturador (Figura 4):

1. Após o operador dar ordem de início de um novo ciclo (premindo o botão Ciclo), é fechada a válvula V7, são abertas as válvulas V1, V3 e V5, e é acionada a Bomba.
2. Quando a quantidade B1 (B2) de produto A (B) para dissolver for depositada no Prato1 (2), é fechada a válvula V1 (V3). *Atenda a que a que os sensores dos Pratos (Prato1 e Prato2) podem balançar mas por defesa mecânica, depois do sensor Prato1(2) ter sido ativado, não deverá voltar a ser ativada a válvula respetiva mesmo que o sensor volte a ficar inativo, ie. as válvulas não são reabertas.*
3. Quando o nível máximo (M\_max) do misturador for atingido, a Bomba pára e a válvula V5 é fechada. *Atenda a que a o sensor de nível de líquido é uma “boia” e pode balançar; sempre que M\_Max se desativar, a Bomba deve voltar a funcionar e V5 abrir.*
4. De seguida (com M\_Max e os Pratos ativos) o motor da pá do misturador é acionado, bem como o motor do tapete no sentido esquerdo. Ao mesmo tempo que são abertas as válvulas V2 e V4.
5. Após o intervalo de tempo T1 (3 segundos), o tapete pára e as válvulas V2 e V4 são fechadas.
6. Após o intervalo de tempo T2 (4 segundos), depois de T1, o motor do misturador pára, estando então concluído o processo de mistura.
7. Então, a válvula V7 é aberta para se iniciar o consumo do produto.
8. Quando o produto tiver sido todo consumido (M\_min ativo), o ciclo estará completo e um novo ciclo pode ser iniciado
9. **Especificação Adicional (controlo supervísório):** Considere que o botão Paragem só é ativado quando o misturador se encontra a executar as operações descritas nos pontos 2, 3 e 4 do ciclo de funcionamento normal. Nas outras fases do ciclo, a atuação do botão Paragem deve ser ignorada. *Se o botão Paragem for premido, o Grafcet de controlo normal da misturadora deve congelar, isto é, todas as etapas e saídas devem permanecer inalteradas. Depois do botão Paragem ser desativado, o Grafcet é reinicializado.*

## 5. Relatório final de cada aula

---

Até ao momento definido no Moodle para cada uma das semanas, submeter os 2 seguintes elementos:

- Um “relatório”, documento de processador de texto (word ou outro), com os seguintes elementos:
  - *Título:* S&A - Lab12\_Aula1 ou Lab12\_Aula2
  - *Turma:* ...
  - *Bancada:* ...
  - *Data e hora:* ...
  - *Autor(es):* ...
  - Incluir a foto/digitalização do(s) grafcet(s) implementados na aula em causa
  - Breve descrição das funcionalidades conseguidas (tópicos), incluindo debug
  - Fazer Copy and Paste de todo o ficheiro “main.c”
- O Zip de toda a diretoria SimpleCSim (incluindo o executável final); *NÃO incluir o DevCPP*; submeter um ficheiro com o nome:  
**Lab12\_Aulax\_Tyy\_Bzz\_PrimNomeUltNomeAAA+PrimNomeUltNomeBBB.ZIP**

- Fim do Guião do TP, continua com anexos -

## Anexo I:

---

São fornecidos alguns recursos:

- `cur_time` - variável global que conta o tempo em décimos de segundo utilizando uma variável `timer0`
- tipo de variáveis `ms_timer` - temporizador em milissegundos, ie., utilizar para contagens de tempo; por exemplo para uma variável `ms_timer timer1`; lançar com `start_timer(&timer1, 1000)`; e consultar com `get_timer(&timer1)`;
- para debug, utilizar GOTOXY que coloca o cursor da consola e depois `printf()`.

### Exemplo de código #1:

```
ms_timer timer1; // Declare global variable of type ms_timer
int main()
{
    initialize_all();
    start_timer(&timer1, 1000); // initialize timer for first time (mili secs)

    while (1) { // infinite cycle
        read_all_inputs(); // global variables "in.*" is updated from <<buttons>>

        //// Start of test code
        if (in.b0) printf("."); // if button 1, then print
        if (in.b2) out.b2=1-out.b2; // if button 2, then toggle out.b2

        if (get_timer(&timer1)) { // If timer expired
            start_timer(&timer1, 1000); // Re-Start it
            out.b7=1-out.b7; // and toggle out.b2
        }

        out.b1=(cur_time & 1); // make led blink 0.1 sec on, 0.1 sec off; uses "timer0"
        //// End of test code

        // **** Inserir código aqui / Insert code here ****

        fflush(stdout);
        write_all_outputs(); // write global variables "out.*" to the <<leds>>
    }
}
```

### Trecho de código exemplo #2:

```
#define ESQUERDA out.b0
#define BOMBA_V5 out.b1
#define MOTOR_PA out.b2
#define V1 out.b3
#define V2 out.b4
#define V3 out.b5
#define V4 out.b6
#define V7 out.b7

#define M_MAX in.b0
#define M_MIN in.b1
#define PRATO1 in.b2
#define PRATO2 in.b3
#define PARAGEM in.b4
#define CICLO in.b5

GOTOXY(1,3);
printf("Entradas: M_MAX=%01d, M_MIN=%01d, PRATO1=%01d, PRATO2=%01d, PARAG=%01d, CICLO=%01d\n\r",
        M_MAX, M_MIN, PRATO1, PRATO2, PARAGEM, CICLO);

GOTOXY(1,5);
printf("Saidas: ESQ=%01d B_V5=%01d MOT_PA=%01d V1=%01d V2=%01d V3=%01d V4=%01d V7=%01d\n\r",
        ESQUERDA, BOMBA_V5, MOTOR_PA, V1, V2, V3, V4, V7);
```

- Fim do guião -