

Introdução

Este anexo, proporciona-lhe um primeiro contacto com a programação de autómatos através da linguagem Grafcet.

O Anexo II inclui elementos para comando hierárquico.

Nesta UC, apenas será utilizada a linguagem ST.

Na primeira parte deste anexo são apresentados o editor, os objetos gráficos e a programação de ações e transições.

Na segunda parte, vai desenvolver uma pequena aplicação.

Nota:

Determinados modelos de Autómatos programáveis permitem programação directamente em Grafcet (como é o caso dos autómatos utilizados nas aulas da disciplina), mas em outros casos o fabricante não disponibiliza tal ferramenta.

Mesmo que a ferramenta não esteja disponível, o Grafcet é uma importante ferramenta conceptual que deve ser utilizada sempre que possível para estrutura o código. De facto, é simples compilar manualmente o código de um modelo Grafcet em texto estruturado e, dessa forma, obter código fácil de documentar e manter em projetos de dimensão considerável.

Introdução ao Editor gráfico

O diagrama é construído utilizando o editor gráfico que se encontra na figura seguinte.

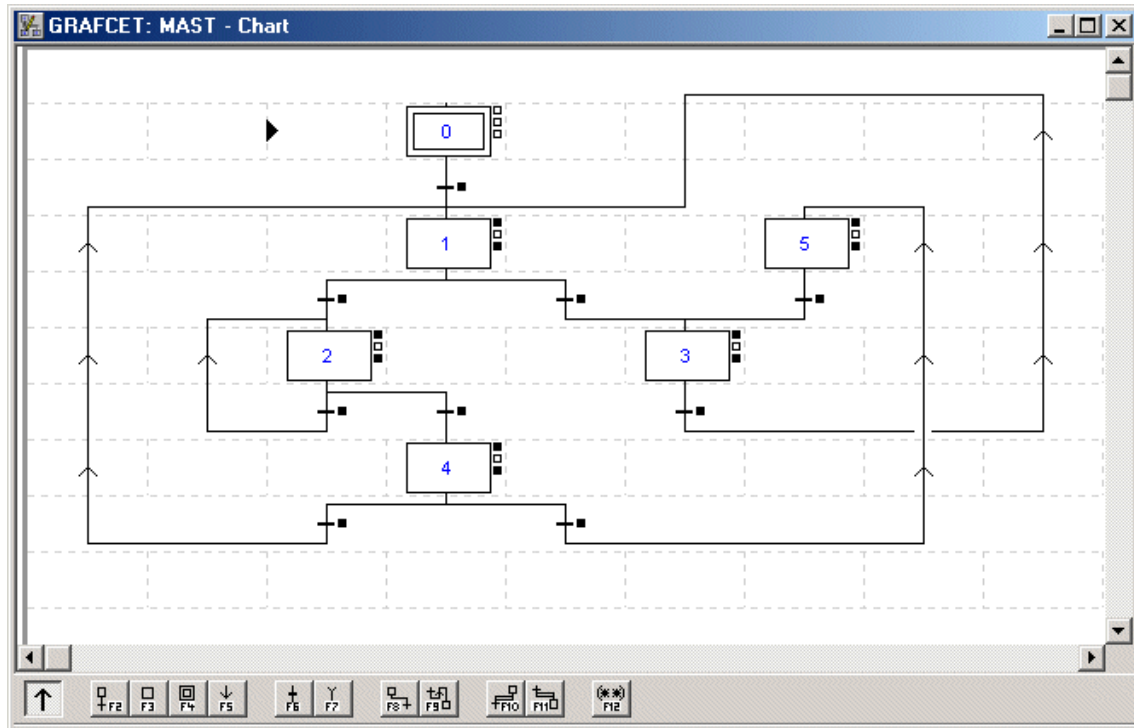


Figura 1 - editor de GR7 do PL7 Junior

O editor suporta 8 páginas, referenciadas de 0 a 7 na barra de estado. Cada página apresenta a forma de uma matriz com 14 linhas e 11 colunas que definem 154 células. Uma mesma página pode conter vários diagramas. A página está dividida em grelha e apenas pode ser colocado um objeto - etapa, transição, etc. - por elemento da grelha.

Cada página do editor tem dois tipos de linhas:

- Linhas de etapas - onde podem ser colocadas etapas, macro-etapas¹ e conectores.
- Linhas de transição - onde podem ser colocadas transições e conectores fonte.

Os comentários são objectos independentes, que não estão associados a etapas ou transições, que podem ser introduzidos em qualquer um dos tipos de linha.

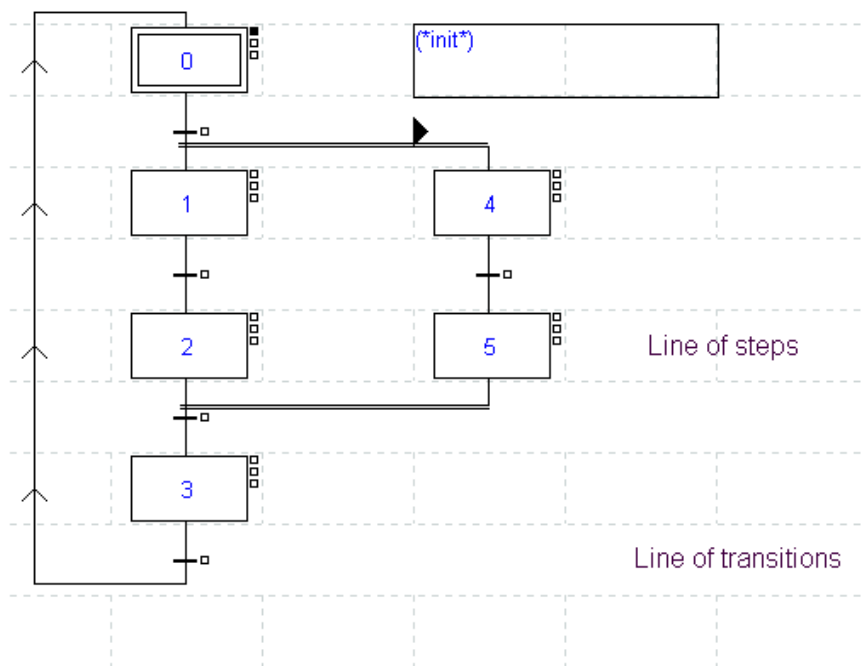



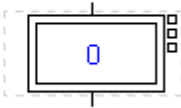


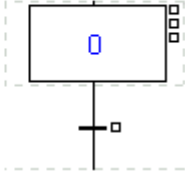
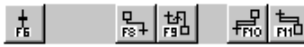











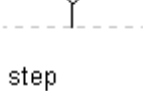


Figura 2 - página exemplo de diagrama Grafcet simples

Objectos gráficos

Os objectos gráficos que podem ser utilizados para construir os diagramas (apresentados da Figura 3 à Figura 8) são os seguintes:

- Etapas (iniciais ou simples)
 - Pode-se associar acções programadas em LD, ST ou IL
- Etapa + transição.
- Transições
 - A condição associada a cada transição (receptividade) pode ser programada em LD, ST ou IL
- Arcos (orientados).
- Conectores.
- Comentários.

¹ Não suportado pelo TSX 37-21/22

<p>Steps</p>  <p>Initial F4</p>  <p>Single F3</p>  <p>Figura 3 - etapas</p>	<p>Step + Transition</p>  <p>F2</p>  <p>Figura 4 - etapa + transição</p>
<p>Transitions</p>  <p>Single F6</p>  <p>AND divergence F11</p>  <p>AND convergence F10</p>  <p>Transition -> Step link F8</p>  <p>Transition -> Step link F9</p>  <p>Figura 5 - transições</p>	<p>Directed links</p>  <p>Upwards</p>  <p>Downwards</p>  <p>To the left or right</p>  <p>Figura 6 - arcos orientados</p> <p>Connectors</p>  <p>Destination F5</p>  <p>Source F7</p>  <p>n = destination step number n = source step number</p> <p>Figura 7 - transições</p>
<p>Comments</p>  <p>Figura 8 - comentários</p>	
 <p>Figura 9 - totalidade da barra de ferramentas do editor</p>	

Objectos específicos do GR7

Existe um conjunto de objectos que são específicos da programação em GR7:

- Objectos (*bits*) associados às etapas - %Xi.
- Objectos (*bits*) do sistema associados ao GR7 - %S21, %S22, %S23 e %S26.
- Objectos (*words*) associados às etapas e que indicam o tempo de actividade das mesmas - %Xi.T.
- Objectos (*words*) do sistema associadas ao GR7 - %SW20 e %SW21.

Os objectos associados às etapas - %Xi - tomam o valor 1 quando a respectiva etapa está activa. O seu valor pode ser testado em todas as tarefas de processamento.

Os objectos que indicam o tempo de actividade das etapas - %Xi.T - são incrementados de 100 em 100 *ms* e apresentam há quanto tempo a etapa está activa em (*ms*). Quando uma etapa é activada, o conteúdo deste objecto é colocado a zero e passa a ser incrementado enquanto esta se mantiver activa. Quando uma etapa é desactivada o seu conteúdo é mantido. A menos de necessidade imperiosa, utilize sempre estes temporizadores *standard* do Grafcet.

Acções associadas com etapas

Cada etapa pode ter acções associadas (programadas em LD, ST ou IL) que apenas são executadas enquanto a etapa a que se encontram associadas estiver activa. São admissíveis três tipos de acções que podem ser utilizadas em simultâneo numa mesma etapa:

- Acção na activação - executadas assim que a etapa a que estão associadas se torna activa (e apenas nessa altura).
- Acções contínuas - executadas continuamente enquanto a etapa a que estão associadas se encontra activa.
- Acções na desactivação - executadas quando a etapa a que estão associadas é desactivada (e apenas nessa altura).

As regras de programação das acções são as seguintes:

- Uma acção associada a uma etapa deve ser activada aquando da activação da etapa e desactivada aquando da desactivação da mesma etapa
- Uma acção que afecta várias etapas deve ser activada na activação da etapa onde a acção se inicia e desactivada na desactivação da etapa onde a acção termina
- É possível ter acções condicionais, isto é, é possível codificar condições de activação das acções.

Condições de transição (receptividade)

Cada transição tem associada condições de transição (programada em LD, ST ou IL) que apenas são avaliadas quando a transição a que estão associadas está validada (receptiva). No **PL7 Junior** uma condição de transição não programada é sempre avaliada como falsa.

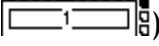
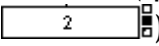
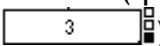


Quando as condições de transição são programadas em ST é necessário considerar algumas diferenças para a utilização normal da linguagem:

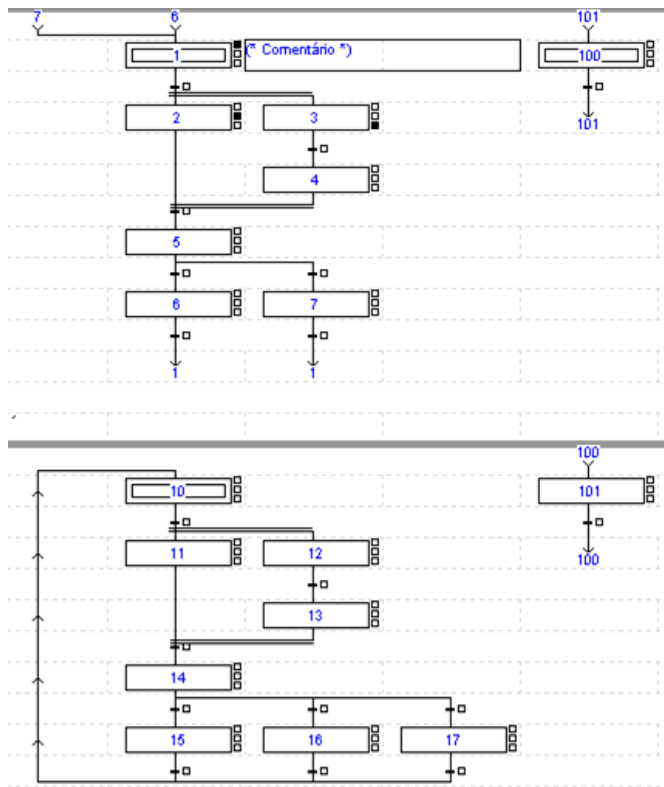
- Não é utilizada a etiqueta %L
- Não podem ser utilizadas instruções de acção (exemplo: atribuições a bits)
- Não são permitidos saltos nem chamadas a subrotinas
- Não se termina a receptividade com “;”

Como consequência, o código da transição deve poder ser avaliado tal como uma expressão que resulte globalmente em `True` ou `False` (tal como por exemplo a condição depois do `IF` e antes do `THEN`).

Funcionalidades adicionais do editor

Relativamente à figura ao lado, confira os seguintes dados:

- Etapas 1, 10 e 100 são iniciais
- A etapa 1 tem a acção à activação definida (quadrado superior cheio )
- A etapa 2 tem a acção ao nível definida (quadrado do meio cheio )
- A etapa 3 tem a acção à desactivação definida (quadrado inferior a cheio )
- À saída da etapa 100 está uma transição que depois liga à etapa 101 (ferramenta )
- À entrada da etapa 100 está uma transição proveniente da etapa 101 (ferramenta )
- As ligações anteriores conferem:
 - da etapa 100 liga-se à etapa 101 que recebe esta ligação
 - da etapa 101 liga-se à etapa 100 que recebe esta ligação
- É possível utilizar setas para ligar dentro ou entre páginas de Grafcet



A figura seguinte procura ainda documentar outras situações:

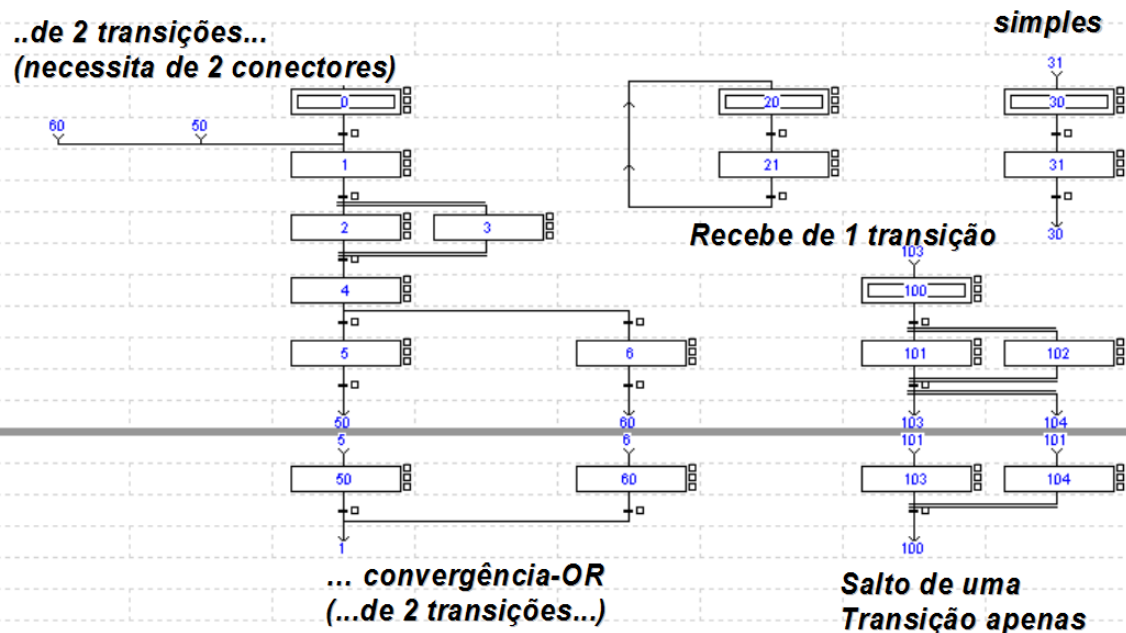


Figura 10 - Exemplo de utilização de conectores em situações complexas

Criação de aplicações Grafcet

Para criar uma aplicação em Grafcet é necessário activar esta opção durante a criação do projecto, como se mostra na figura seguinte.

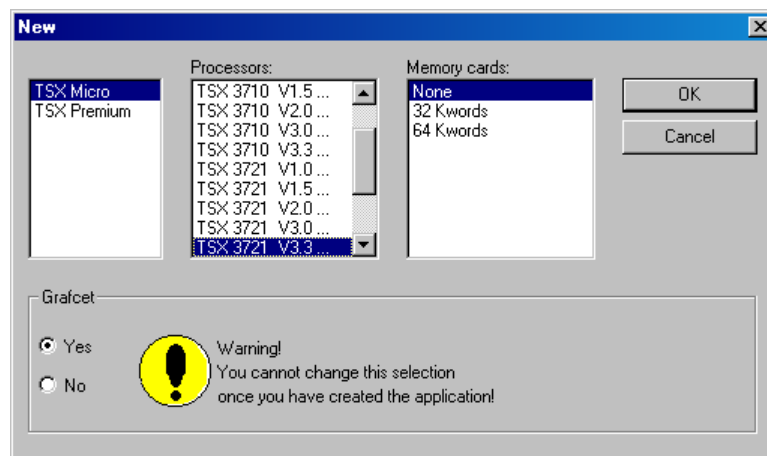


Figura 11 Criação de nova aplicação no PL7

De seguida, e como anteriormente, é necessário configurar o hardware do automático. Para isso, na janela *Application Browser* seleccione *STATION/ Configuration/ Hardware Configuration* e adicione o módulo TSX DMZ 28 DR na posição adequada.

Confirme a alteração no botão .

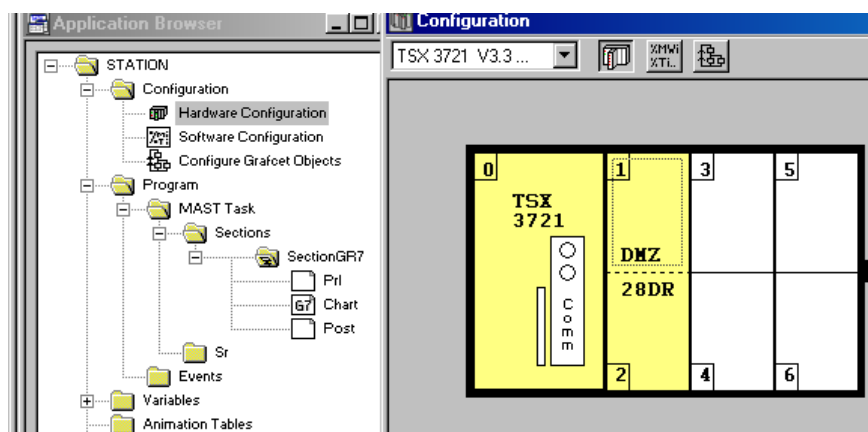



Figura 12 Configuração do Hardware

Tal como nos trabalhos práticos anteriores, na janela Application Browser seleccione STATION/ Variables/ I/O:

- No modulo 1: TSX DMZ 28 DR defina as entradas.
- No modulo 2: TSX DMZ 28 DR defina as saídas.

Tal como nos trabalhos práticos anteriores, na janela Application Browser seleccione STATION/ Animation Table/ Create:

- Adicione as variáveis de entrada e de saída à tabela de animação
- Confirme a alteração no botão .

Exemplo de edição de modelo Grafcet

Neste ponto será exemplificada a edição do seguinte modelo Grafcet:

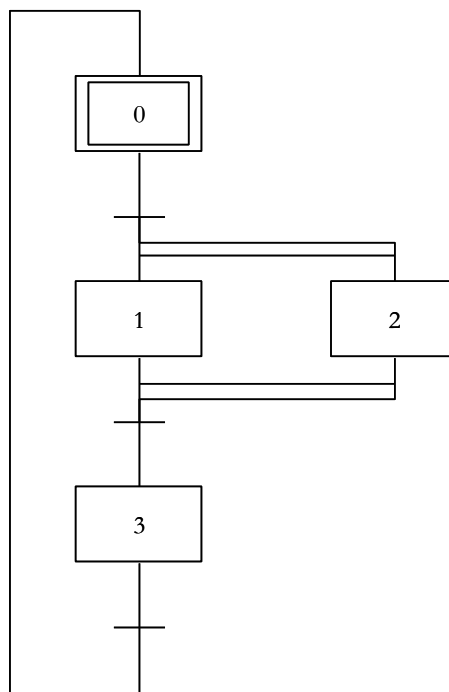


Figura 15 - Diagrama Grafcet de teste

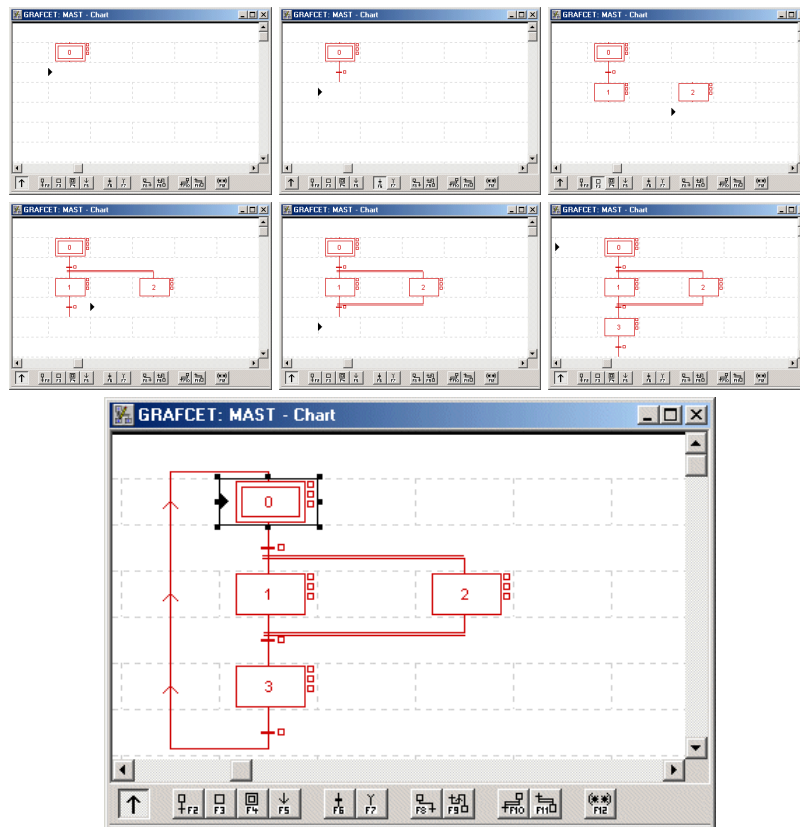










Figura 13 - passos na criação de um diagrama GR7 de teste

Crie o Grafcet de teste apresentado na figura anterior, de acordo com os passos seguintes:

1. Para iniciar a edição, no Application Browser deve-se seleccionar STATION/ Program/ MAST Task/ Sections / Section GR7 / [G7] Chart (ver Figura 12).
2. Selecione uma etapa inicial (F4 ou ) e coloque na segunda célula da segunda linha.
3. Selecione uma transição (F6 ou ) e coloque na linha imediatamente abaixo.
4. Selecione uma etapa (F3 ou ) e coloque na linha imediatamente abaixo. Coloque também uma etapa na mesma linha e duas células para a direita.
5. Para adicionar uma divergência do tipo AND selecione (F11 ou ) , clique sobre a transição e depois sobre a etapa 2.
6. Coloque outra transição na linha abaixo e duas células para a esquerda.
7. Para inserir uma convergência do tipo AND selecione (F10 ou ) , clique na etapa 2 e de seguida sobre a transição que acabou de adicionar.
8. Coloque na linha imediatamente abaixo uma etapa e uma transição (F2 ou ) .
9. Selecione uma ligação transição - etapa (F9 ou ) e ligue a última transição que adicionou à etapa 0.


Após terminar a construção do diagrama deve validá-lo no botão  ou através do menu *Edit/Confirm*. Quando um diagrama é confirmado o seu aspecto altera-se: os objectos passam de vermelho para preto e a delimitação das páginas torna-se cinzenta. Os erros são reportados na barra de estado da aplicação.

Nesta altura podem ser definidas as condições de transição (para as diferentes transições) e programadas as acções (associadas às diferentes etapas).

Para efectuar a programação das acções e das condições de transição o diagrama deve estar validado. Sempre que efectuar uma alteração no diagrama deve voltar a validá-lo.

A programação das transições é acessível seleccionando a transição com o botão do lado direito do rato. Abra a transição e seleccione a linguagem para a programação ST.

Neste exemplo vamos seleccionar a linguagem ST e programar a condição de transição para a transição entre as etapas 0 e 1 (

Figura 14). Após programar a condição deve validá-la através do botão .

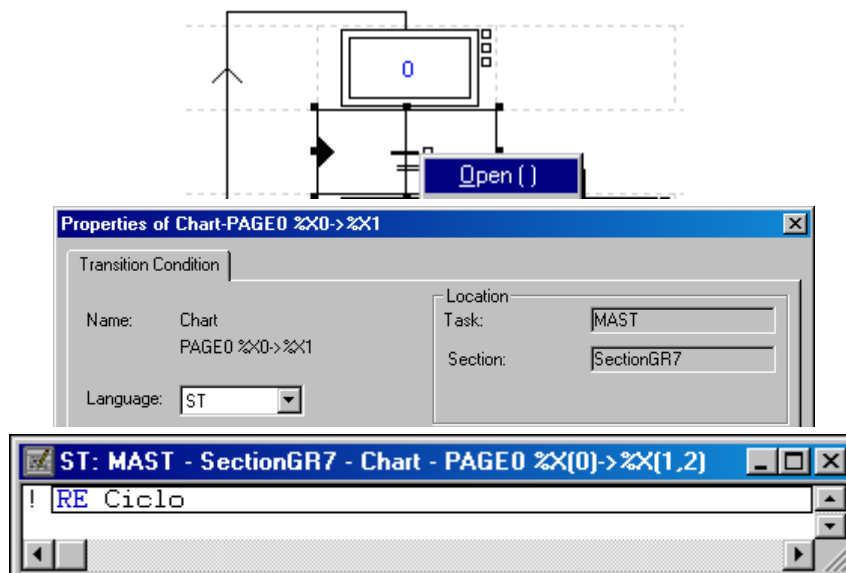
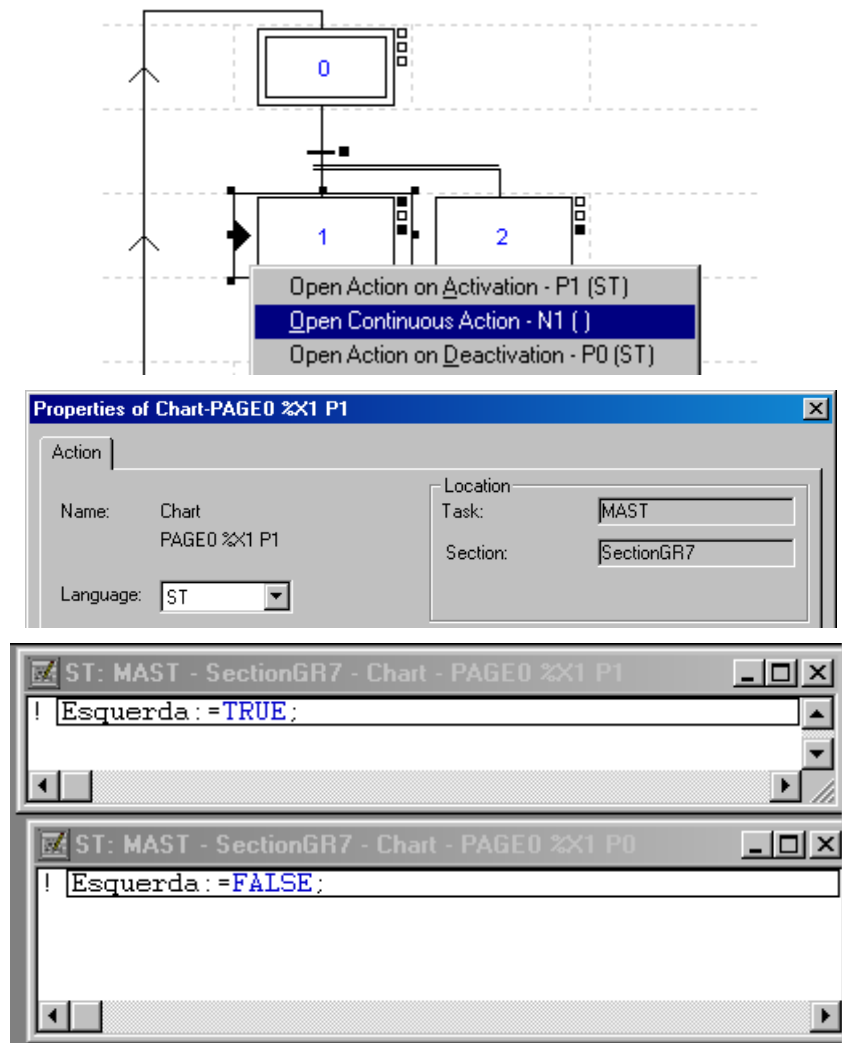


Figura 14 - definição da receptividade associada a uma transição X0->X(1,2)

Fica ao cargo do aluno programar as restantes receptividades com sinais de teste, presentes no kit do trabalho actual.

A programação das acções é efectuada seleccionando a etapa pretendida com o botão direito do rato (ou com o botão esquerdo e de seguida premindo Shift+F10). Comece por seleccionar o tipo de acção associado com a etapa (acção na activação, acção contínua ou acção na desactivação) e de seguida a linguagem em que pretende programar a acção (LD, ST ou IL). Selecione ST.



**Figura 15 - programação das acções:
%X1 P1 à activação e %X1 P0 à desactivação**

Para este caso vão ser utilizadas acções na activação e acções na desactivação, isto é, que são executadas enquanto a etapa a que estão associadas é activada e quando é desactivada, programadas em ST. A Figura anterior apresenta a programação da acção na activação associada à etapa 1. Após programar a acção deve validá-la através do botão . As restantes acções devem ser programadas de modo semelhante.

Fica ao cargo do aluno programar as restantes acções com sinais de teste, presentes no kit do trabalho actual.

No final do trabalho, o seu diagrama deve ter o aspecto mostrado na Figura 14 mas onde os quadrados a cheio junto aos objectos (transições e etapas) indicam quais são os que têm os respectivos elementos de programação definidos.

Animação de execução

Quando o autómato se encontra em modo **Run** é possível visualizar no editor de GR7 a evolução do estado do autómato.

Teste o Grafcet e regras de evolução associadas para dominar o funcionamento do ambiente de desenvolvimento.

Uma etapa activa é indicada a preto enquanto que uma etapa não activa é indicada a branco. Para tal, a função de animação deve estar activa (quando o PLC está no modo **Run**).

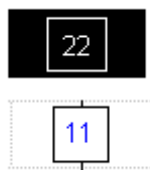


Figura 16 - Animação do GR7

Tal como no TP anterior pode utilizar a tabela de animação para observar o estado actual das variáveis que necessitar, incluindo os objectos especiais do grafcet, exemplo %X1.T.

Anexo II: Estrutura de um programa em Grafcet

Este anexo apenas é relevante para a 2ª aula prática, onde terão de ser implementadas as especificações adicionais com base em macro-acções.

As aplicações Grafcet são estruturadas em três módulos:

- Pré processamento (PRL): executado antes do Grafcef, utilizado para processar a lógica de entrada e inicializar o Grafcet.
- Grafcet (CHART): condições de transição associadas às transições e acções associadas às etapas são programadas nas páginas do Grafcet.
- Pós processamento (POST): executado depois do Grafcet, utilizado para processar a lógica de saída, monitorizar e definir esquemas de segurança.

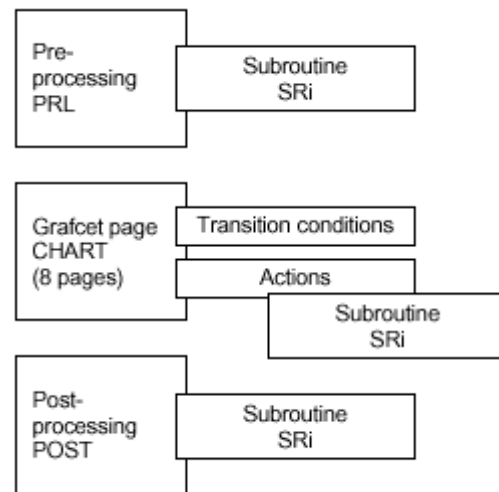


Figura 17

Na primeira parte deste trabalho prático, apenas é necessário programar no módulo CHART.

Já na segunda parte, a programação das macro-acções terá de ser efectuada em texto estruturado no módulo PRL.

Relativamente às macro-acções, recorda-se que, com o software de desenvolvimento disponível no laboratório:

1. é possível actuar directamente sobre o estado de uma dada etapa i do Grafcet através de instruções de Set e Reset sobre o bit X_i associado a essa etapa.
2. é possível actuar sobre o estado global do Grafcet através de instruções de Set e Reset sobre os seguintes bits sistema:
 - **S9:** Todas as saídas em estado de repouso (desligadas)
 - **S21:** para re-inicialização do Grafcet (voltar ao estado inicial)
 - **S22:** desactivação de todas as etapas
 - **S23:** desactivação de todas as transições
3. as instruções de **Set** e **Reset** dos bits anteriores (X_i e S_i) apenas podem ser incluídas no módulo de pré-processamento.
4. quando uma etapa é activada/desactivada através de um dos bits anteriores, as acções definidas para a activação/desactivação dessa etapa não são executadas.

NOTA: Algumas destas variáveis são iniciadas todos os ciclos a FALSE (outras não...)

- Fim-