

ANEXO - FEUPAUTOM GRAFCET HIERÁRQUICO

(versão 14-04-2018)

O FEUPAutom, implementa Grafcet hierárquico mas com diversas limitações e restrições.

Para além das variáveis Xi e Xi_T, há dois outros mecanismos para implementação de hierarquia (com vista a controlo supervisão).

Um primeiro mecanismo diz respeito à produção de código ser feita por página de Grafcet. Desta forma, todo o código da página 3 é produzido antes de todo o código da página 2 e etc. até ao código associado ao Grafcet da página 0.

O outro mecanismo diz respeito à possibilidade de interferir em pontos chave da produção automatizada de código.

O compilador NORMAL do FEUPAutomGrafcet segue a estratégia síncrona delineada na figura 2 (resumidamente lidar com etapas iniciais, desligar etapas a montante de transições disparadas, ligar etapas a jusante de transições disparadas, lidar com temporizações e ações).

Foram então acrescentados os seguintes “hooks”, zonas de código em que o programador pode “interferir” com a execução do Grafcet:

1. **“Zone1”**
 - Se Boot Então Ativa Etapas Iniciais
2. **“Zone2”**
 - Se não Boot Então calcula Transições Disparadas (TrD)
3. **“Zone3”**
 - Para cada TrD, desliga Etapas de a montante
4. **“Zone4”**
 - Para cada TrD, liga Etapas a jusante e faz Reset ao tempo da etapa
5. **“Zone5”**
 - Se página 3 de código Então desliga todas as saídas
6. **“Zone6”**
 - A cada décima de segundo, incrementa os contadores de tempo das etapas ativas
7. **“Zone7”**
 - Executa o código associado a etapas ativas
8. **“Zone8”**

Este código de “zona” é introduzido em “Etapas” com nomes especiais, desde “Zone1” até “Zone8” - ver figura 3.

Atenção:

Se necessitar de bits de memória auxiliar, utilize por exemplo m100 em diante

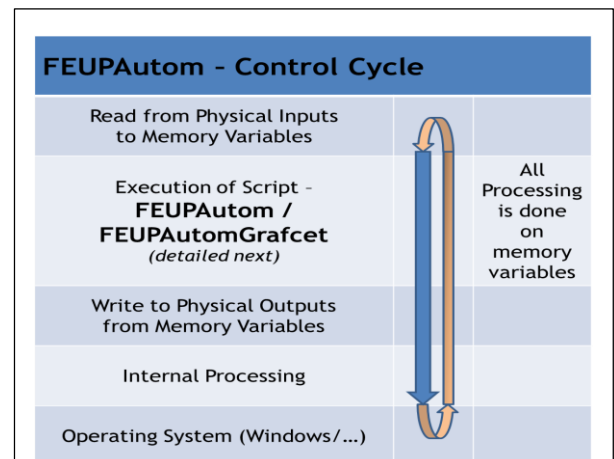


Figura 1 - Ciclo controlo

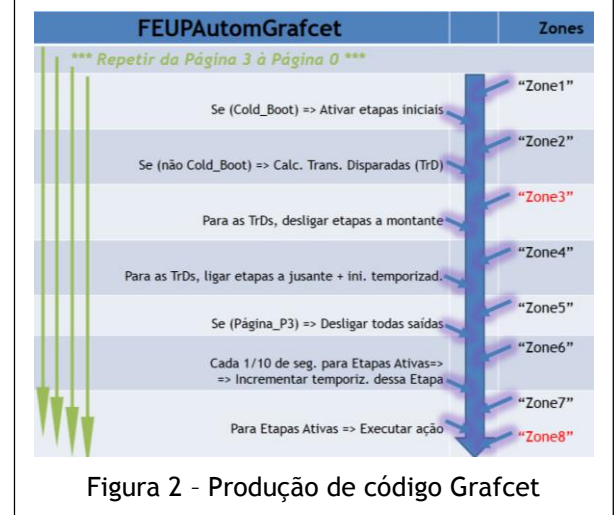


Figura 2 - Produção de código Grafcet

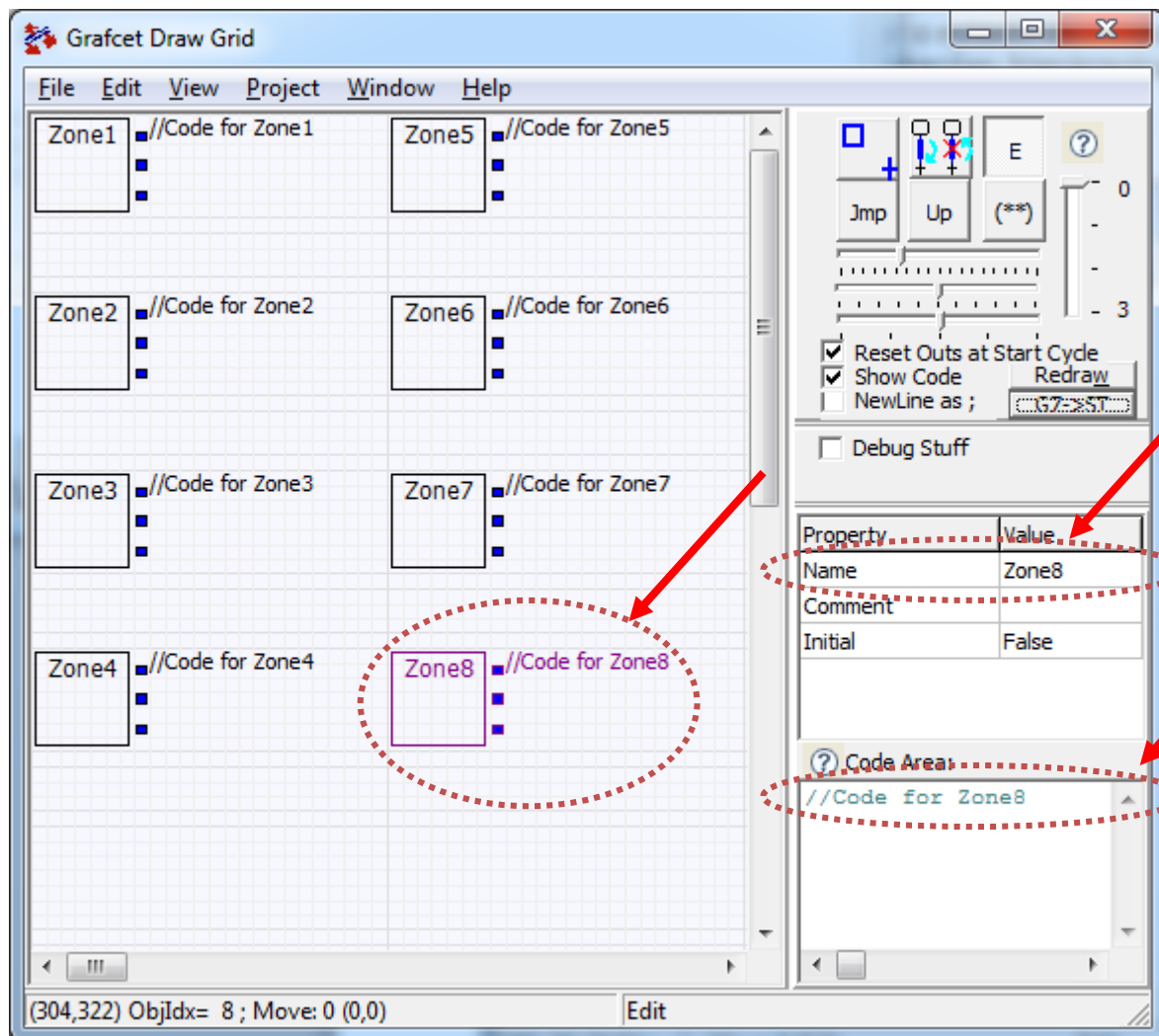


Figura 3 - Página 0 com etapas “zoneX”

Para facilidade de visualização, a figura 3 mostra todas as zonas possíveis mas todas elas são opcionais e cada uma delas pode existir ou não.

Ainda para facilidade de visualização, também opcionalmente, estas etapas estão na página 0 do Grafcet e o código de funcionamento “normal” está na página 1, ver figura 4.

Exemplo de aplicação:

Refira-se ao grafcet exemplo da figura 4.

Inserindo uma etapa com o nome “Zone3” (não mostrada na figura 4) com o código

```
if paragem then    t3:=false; t4:=false;    end_if;
```

fará com que sempre que “paragem” estiver ativo, as transições t3 e t4 sejam sempre falsas, equivalente a congelar (impedir evolução) da parte do grafcet envolvida (etapas 2 e 3), isto para um grafcet com uma página apenas.

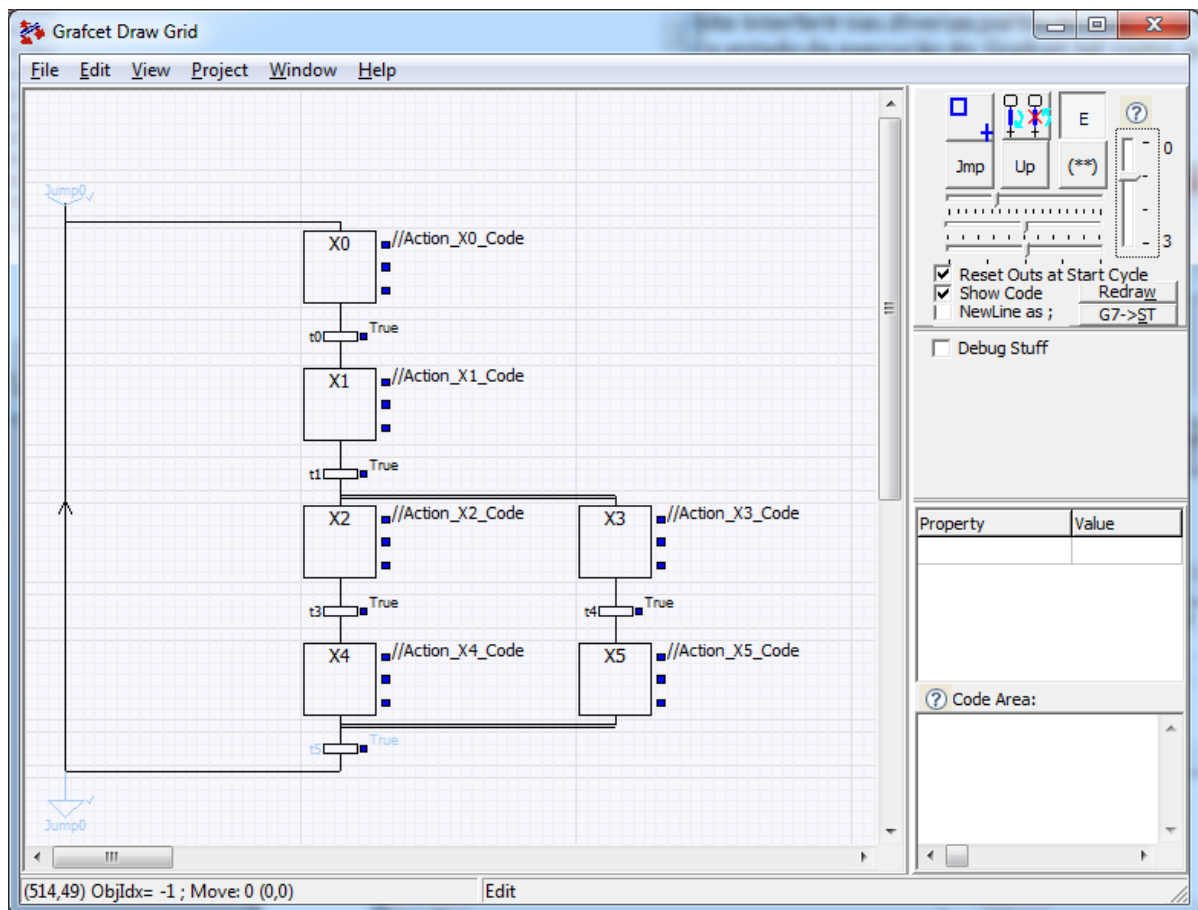


Figura 4 - Grafcet exemplo (ciclo normal de funcionamento) na página 1

O código gerado será então (mostra-se apenas trechos de código):

```

////////////////////////////////////
/// FEUPAutom v3.5 -
/// Code Automatically Generated
////////////////////////////////////

////////////////////////////////////
//////////////////////////////////// Zone1 //////////////////////////////////
////////////////////////////////////

//Code for Zone1

////////////////////////////////////
////////// If boot => Set Initial Steps //////////
////////////////////////////////////

(...)

```

```

////////////////////////////////////
//////////////////////////////////// Zone2 //////////////////////////////////
////////////////////////////////////

//Code for Zone2

////////////////////////////////////
////////// Calc Fired Transitions //////////
////////////////////////////////////

// ObjIdx=10 => Transition "t0"
/ Steps Above: id=9 => X0 ;
/ Steps Below: id=11 => X1 ;
t0 := X0 AND True;

(...)

```

```

////////////////////
//////////////////// Zone3 ///////////////////
////////////////////

//Code for Zone3

////////////////////
////////// ReSet Steps Above fired Tr //////////
////////////////////

// ObjIdx=10 => Transition "t0"
// Steps Above: id=9 => X0 ;
// Steps Below: id=11 => X1 ;
If (t0) Then
    X0:=False;
End_If;

(...)

////////////////////
//////////////////// Zone4 ///////////////////
////////////////////

//Code for Zone4

////////////////////
////////// Set Steps below fired Tr //////////
////////////////////

// ObjIdx=10 => Transition "t0"
// Steps Above: id=9 => X0 ;
// Steps Below: id=11 => X1 ;
If (t0) Then
    X1 := True;
    X1_T := 0;
End_If;

(...)

////////////////////
//////////////////// Zone5 ///////////////////
////////////////////

//Code for Zone5

////////////////////
////////// Unset all Outputs //////////
////////////////////

Q0:=False;
Q1:=False;
(...)
Q47:=False;

```

```

////////////////////
//////////////////// Zone6 ///////////////////
////////////////////

//Code for Zone6

////////////////////
///// If step active increment MW timer of step @ %s16
/////
////////////////////

// ObjIdx=9 => Step "X0"
If (%s16) and (X0) Then X0_T := X0_T+1; end_if;

(...)

////////////////////
//////////////////// Zone7 ///////////////////
////////////////////

//Code for Zone7

////////////////////
///// If step active, execute its action code /////
////////////////////

// ObjIdx=1 => Step "Zone1" (code...)
If Zone1 Then
    //Code for Zone1
End_If;

(...)

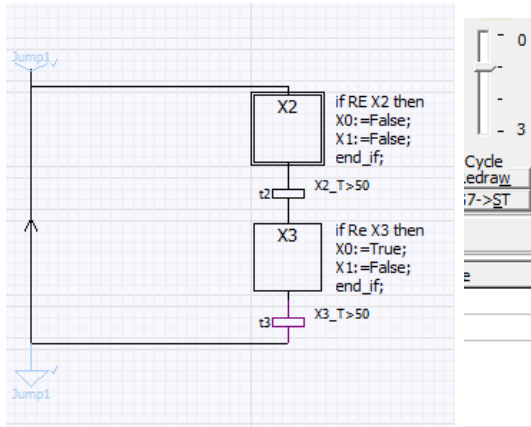
////////////////////
//////////////////// Zone8 ///////////////////
////////////////////

//Code for Zone8

```

Exemplo de aplicação Grafcet Hierárquico

O código associado ao grafcet da esquerda (página 1) aparece antes do outro.



```

////////////////////////////////////
// FEUPAutom v3.6 - Code Automatically Generated:06/05/2014 10:30:59
////////////////////////////////////

//##### Page 3 #####

//////////////// If boot => Set Initial Steps //////////////////

//////////////// Calc Fired Transitions //////////////////

//////////////// ReSet Steps Above fired Tr //////////////////

//////////////// Set Steps below fired Tr //////////////////

//////////////// Unset all Outputs (once for all pages) //////////

Q0:=False;
...
Q47:=False;

///// If step active increment MW timer of step @ %s16 /////

//////// If step active, execute its action code //////////

//##### Page 2 #####

//////////////// If boot => Set Initial Steps //////////////////

//////////////// Calc Fired Transitions //////////////////

//////////////// ReSet Steps Above fired Tr //////////////////

//////////////// Set Steps below fired Tr //////////////////

///// If step active increment MW timer of step @ %s16 /////

//////// If step active, execute its action code //////////

//##### Page 1 #####

//////////////// If boot => Set Initial Steps //////////////////

// ObjIdx=6 => INI_Step "X2"
If (%sw0=0) Then X2 := True; End_If;

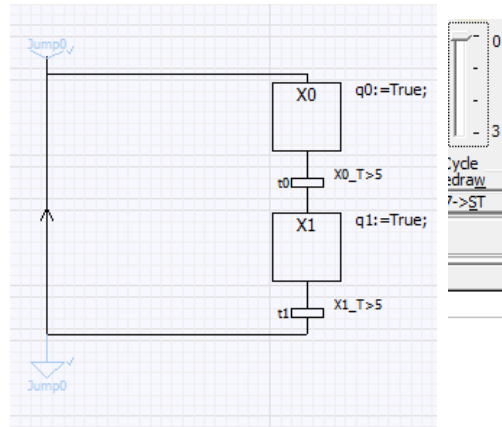
//////////////// Calc Fired Transitions //////////////////

// ObjIdx=7 => Transition "t2"
t2 := X2 AND (X2_T>50) ;
// ObjIdx=9 => Transition "t3"
t3 := X3 AND (X3_T>50) ;

//////////////// ReSet Steps Above fired Tr //////////////////

// ObjIdx=7 => Transition "t2"
If (t2) Then X2:=False; End_If;
// ObjIdx=9 => Transition "t3"
If (t3) Then X3:=False; End_If;

```



```

//////////////////////////////// Set Steps below fired Tr //////////////////////////////////
// ObjIdx=7 => Transition "t2"
If (t2) Then X3 := True; X3_T := 0; End_If;
// ObjIdx=9 => Transition "t3"
If (t3) Then X2 := True; X2_T := 0; End_If;

///// If step active increment MW timer of step @ %s16 /////

// ObjIdx=6 => Step "X2"
If (%s16) and (X2) Then X2_T := X2_T+1; end_if;
// ObjIdx=8 => Step "X3"
If (%s16) and (X3) Then X3_T := X3_T+1; end_if;

//////// If step active, execute its action code //////////

// ObjIdx=6 => Step "X2" (code...)
If X2 Then
  if RE X2 then X0:=False; X1:=False; end_if; End_If;
// ObjIdx=8 => Step "X3" (code...)
If X3 Then
  if Re X3 then X0:=True; X1:=False; end_if; End_If;

//##### Page 0 #####

//////////////// If boot => Set Initial Steps //////////////////

//////////////// Calc Fired Transitions //////////////////

// ObjIdx=1 => Transition "t0"
t0 := X0 AND (X0_T>5) ;
// ObjIdx=3 => Transition "t1"
t1 := X1 AND (X1_T>5) ;

//////////////// ReSet Steps Above fired Tr //////////////////

// ObjIdx=1 => Transition "t0"
If (t0) Then X0:=False; End_If;
// ObjIdx=3 => Transition "t1"
If (t1) Then X1:=False; End_If;

//////////////// Set Steps below fired Tr //////////////////

// ObjIdx=1 => Transition "t0"
If (t0) Then
  X1 := True; X1_T := 0; End_If;
// ObjIdx=3 => Transition "t1"
If (t1) Then
  X0 := True; X0_T := 0; End_If;

///// If step active increment MW timer of step @ %s16 /////

// ObjIdx=0 => Step "X0"
If (%s16) and (X0) Then X0_T := X0_T+1; end_if;
// ObjIdx=2 => Step "X1"
If (%s16) and (X1) Then X1_T := X1_T+1; end_if;

//////// If step active, execute its action code //////////

// ObjIdx=0 => Step "X0" (code...)
If X0 Then q0:=True; End_If;
// ObjIdx=2 => Step "X1" (code...)
If X1 Then q1:=True; End_If;

(***** End of ST Code *****)

```

- Fim do anexo Programação FEUPAutomGrafcet Hierárquico -