“FEUPAutom_Grafcet”

(incluindo Grafcet hierárquico e comparação com norma)

Armando Sousa

27/04/2020

FEUPAutom5
(SoftPLC | Grafcet+ST | ModbusTCP | Cross-Platform)
Code: Armando Jorge Sousa, Bruno Augusto and Paulo Costa

Historico da UC

Sistemas de Máquinas de Estados (SME)

Implementação (Manual) de SMEs
(Arduino, PC e FEUAutom)

Grafcet

Norma IEC 60848

Implementação de Grafcet (FEUPAutom)

Implementação manual de Grafcet sob Arduino e PC

FEUPAutom

- SoftPLC
  - Software que emula um PLC dentro de um PC comum

FEUPAutom

- SoftPLC
  - Software que emula um PLC dentro de um PC comum
- Executa código ST
- Comunica com simulações e dispositivos reais
- *Debugging* fácil
  - Inspeção de variáveis
  - Traçados temporais
  - Em Grafcet mostra etapas ativas

FEUPAutomGrafcet

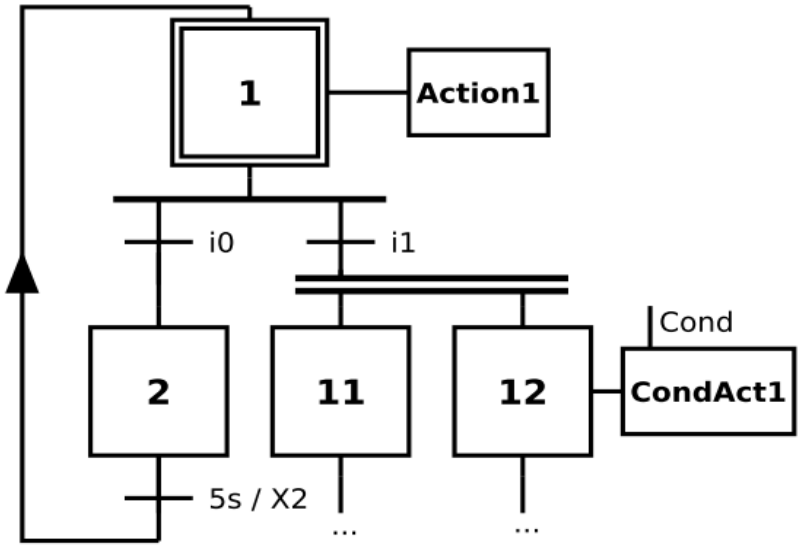
- Editor gráfico de “Grafcet”
- Execução local instantânea com *debug* fácil
- Traduz (*oneway*) de “Grafcet” para ST
  - Útil se não tivermos disponível um compilador
- Na versão >=4, compila para C (sob AVRGCC)

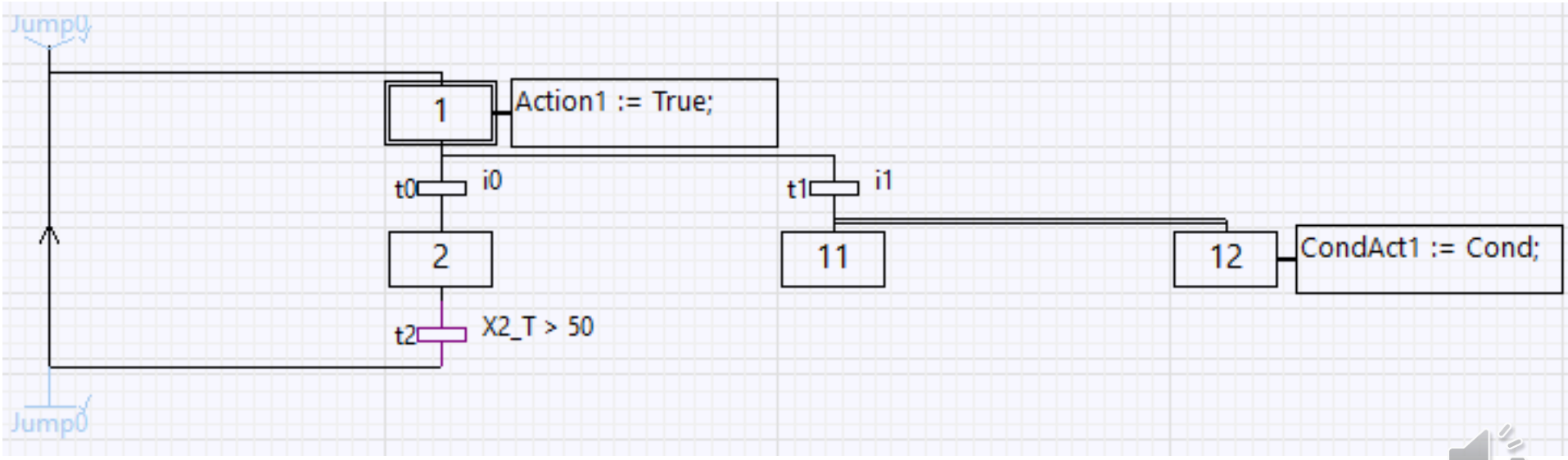
- Utiliza o algoritmo Síncrono –
  – com adaptações
  - daí “FEUPAutomGrafcet”

FEUPAutomGrafcet

Norma => Declarativa

FEUPAutom => Imperativa





Alg. Genérico Síncrono

2. Se Início (Cold_Boot) =>
   => Ativar etapas iniciais

3. Calcular
   Transições Disparadas (TrD)

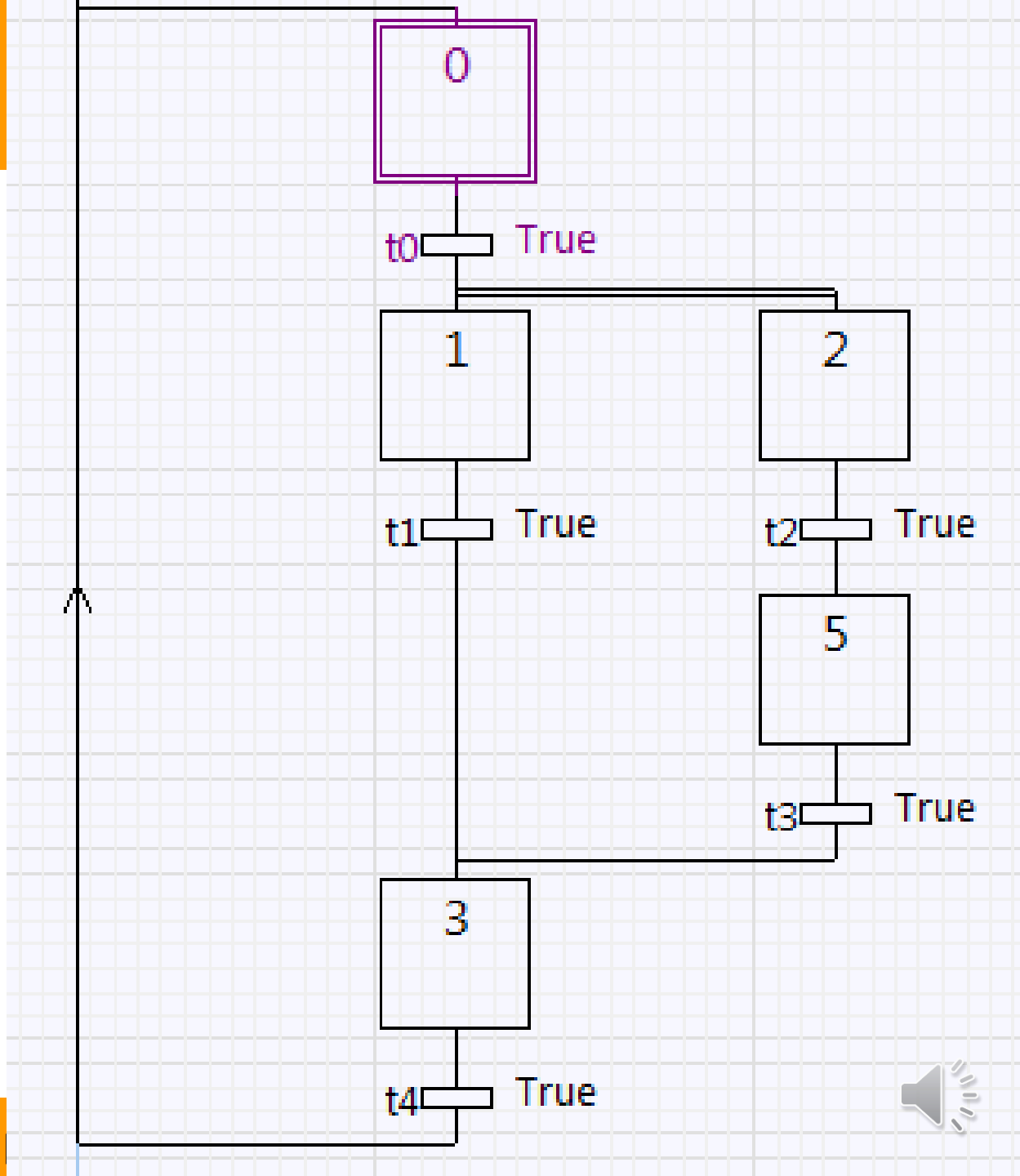
4. Para as TrDs,
   desligar etapas a montante

5. Para as TrDs,
   ligar etapas a jusante

8. Ativar as saídas de
   acordo com as Etapas Ativas

Alg. Genérico Síncrono	FEUPAutomGrafcet
	*** *Para cada Página, da P3 para a P0 (tendo em conta os Hooks)*
2. Se Início (Cold_Boot) => => Ativar etapas iniciais	2.Se (Cold_Boot) => => Ativar etapas iniciais
3. Calcular Transições Disparadas (TrD)	3. Se (não Cold_Boot) => => Calc. Transições Disparadas (TrD)
4. Para as TrDs, desligar etapas a montante	4. Para as TrDs, desligar etapas a montante
5. Para as TrDs, ligar etapas a jusante	5. Para as TrDs, ligar etapas a jusante + ini. temporizad.
	6. (Opcionalmente) Desligar todas saídas
	7. Cada 1/10 de seg. para Etapas Ativas=> => Incrementar temporiz. dessa Etapa
8. Ativar as saídas de acordo com as Etapas Ativas	8. Para Etapas Ativas => Executar ação

Caso de Estudo

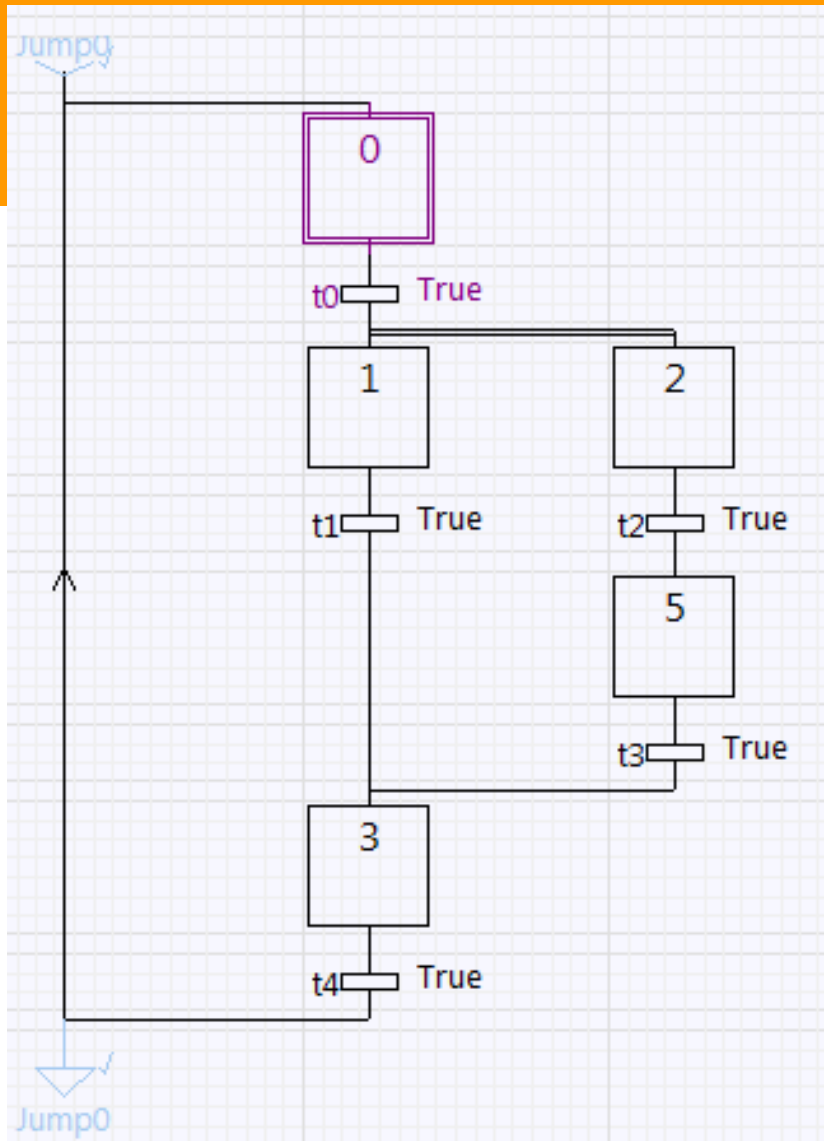


Evolução temporal
(saudável)

Evolução Temporal exemplo:
(reset)

- {0} => disparando t0 =>
- {1,2} => disparando t3+t5 =>
- {3,5} => disparando t2+t4 =>
- {0,3} => disparando t0+t4 =>
- {0,1,2} ...

(até é possível todas
as etapas ficarem ativas!)

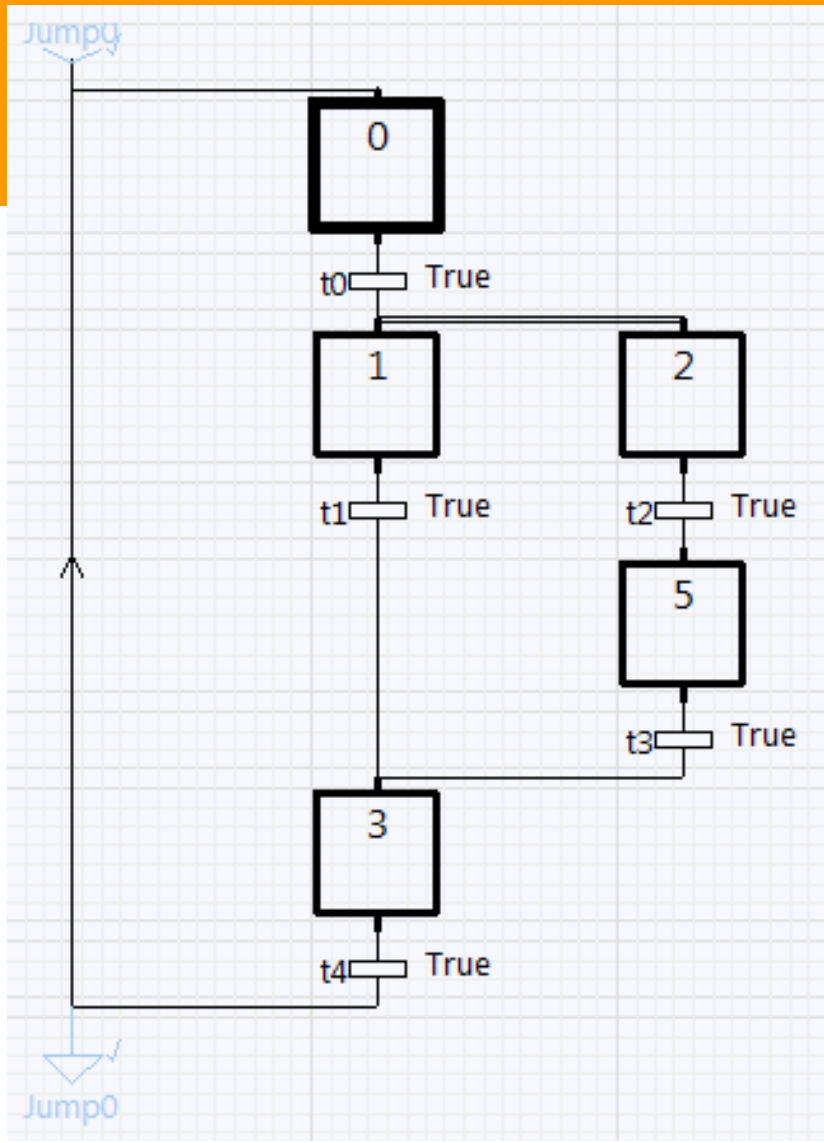


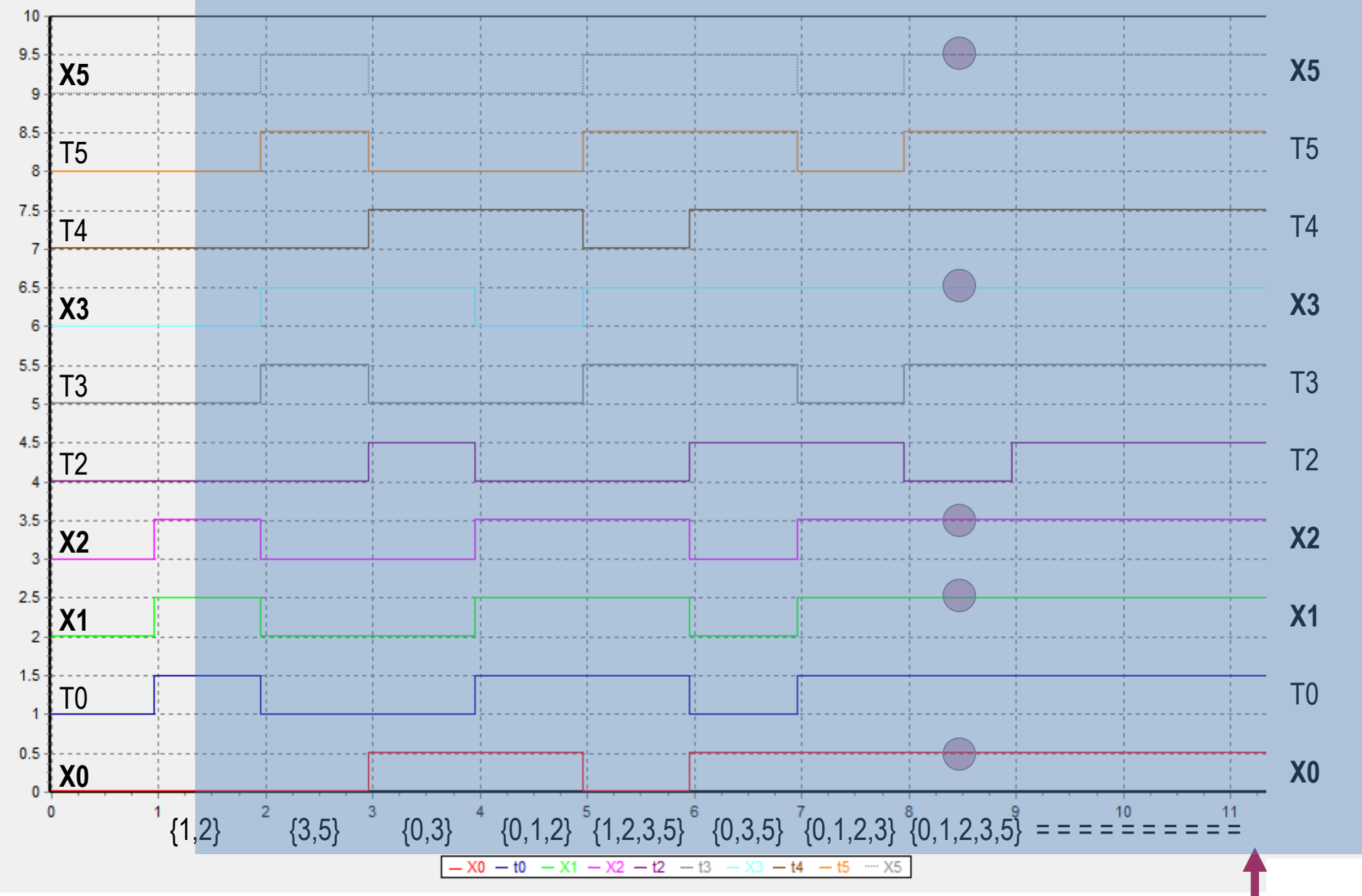
Evolução temporal
(saudável)

Evolução Temporal exemplo:
(reset)

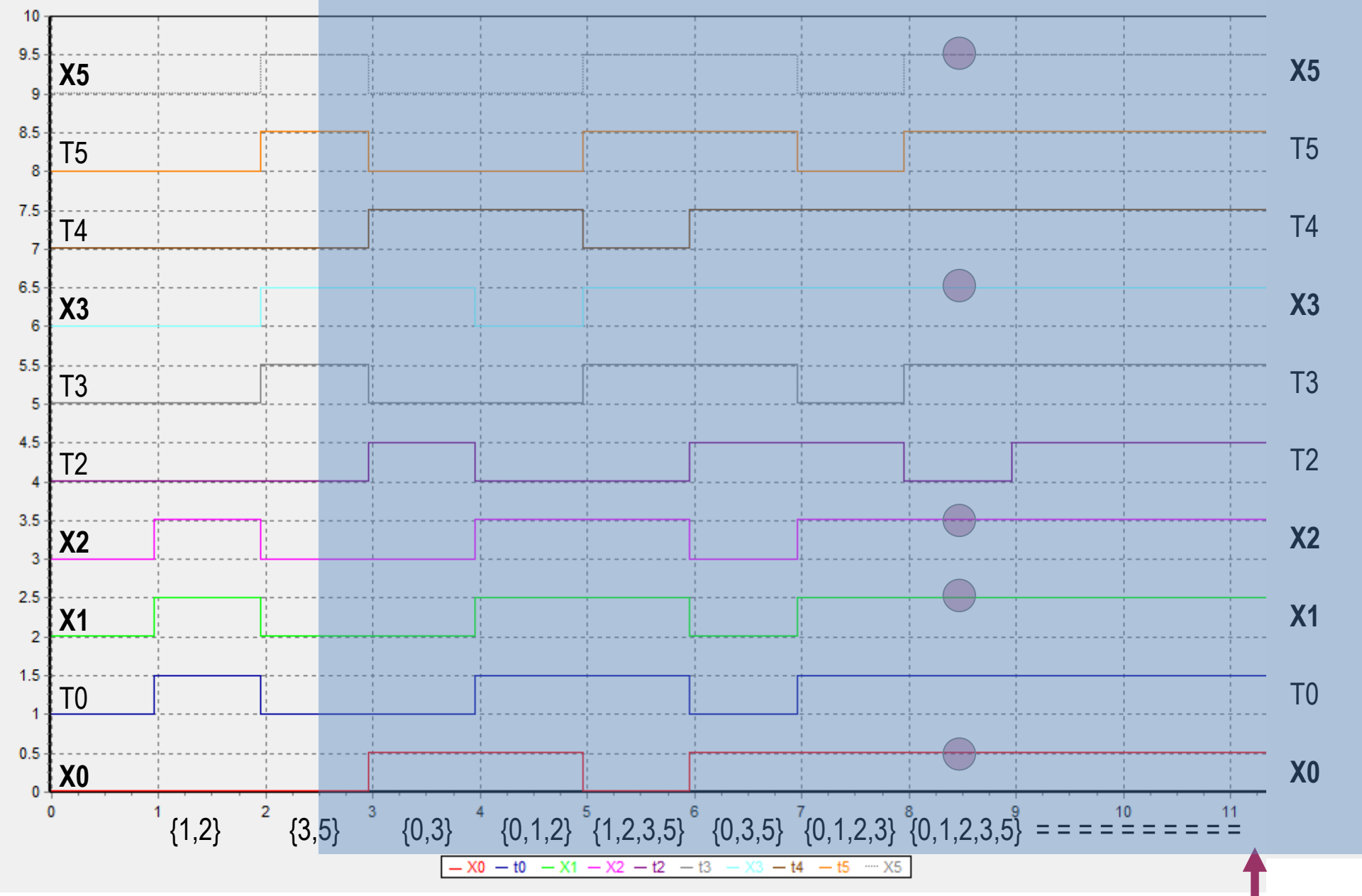
- {0} => disparando t0 =>
- {1,2} => disparando t3+t5 =>
- {3,5} => disparando t2+t4 =>
- {0,3} => disparando t0+t4 =>
- {0,1,2} ...

(até é possível todas
as etapas ficarem ativas!)

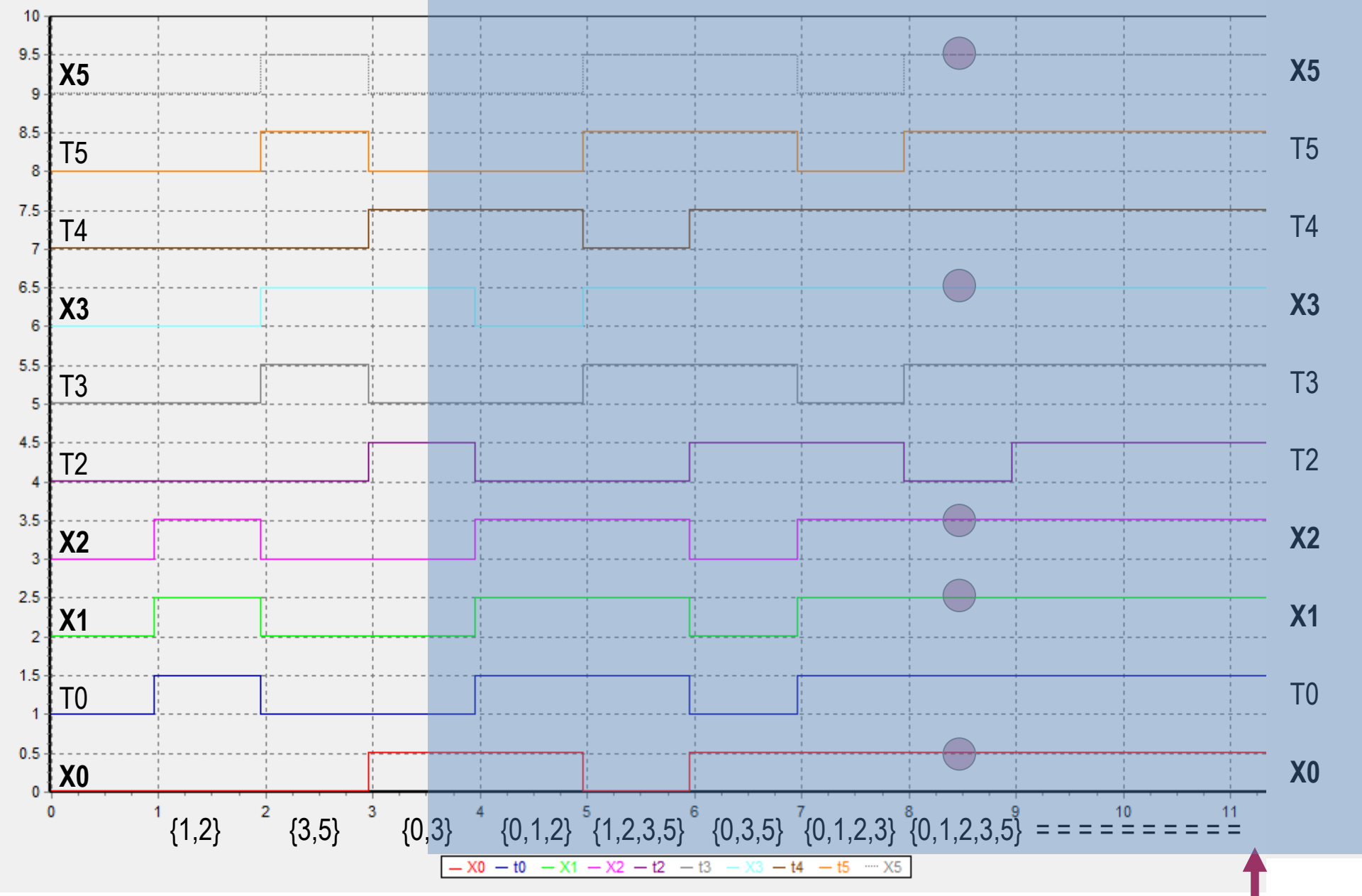




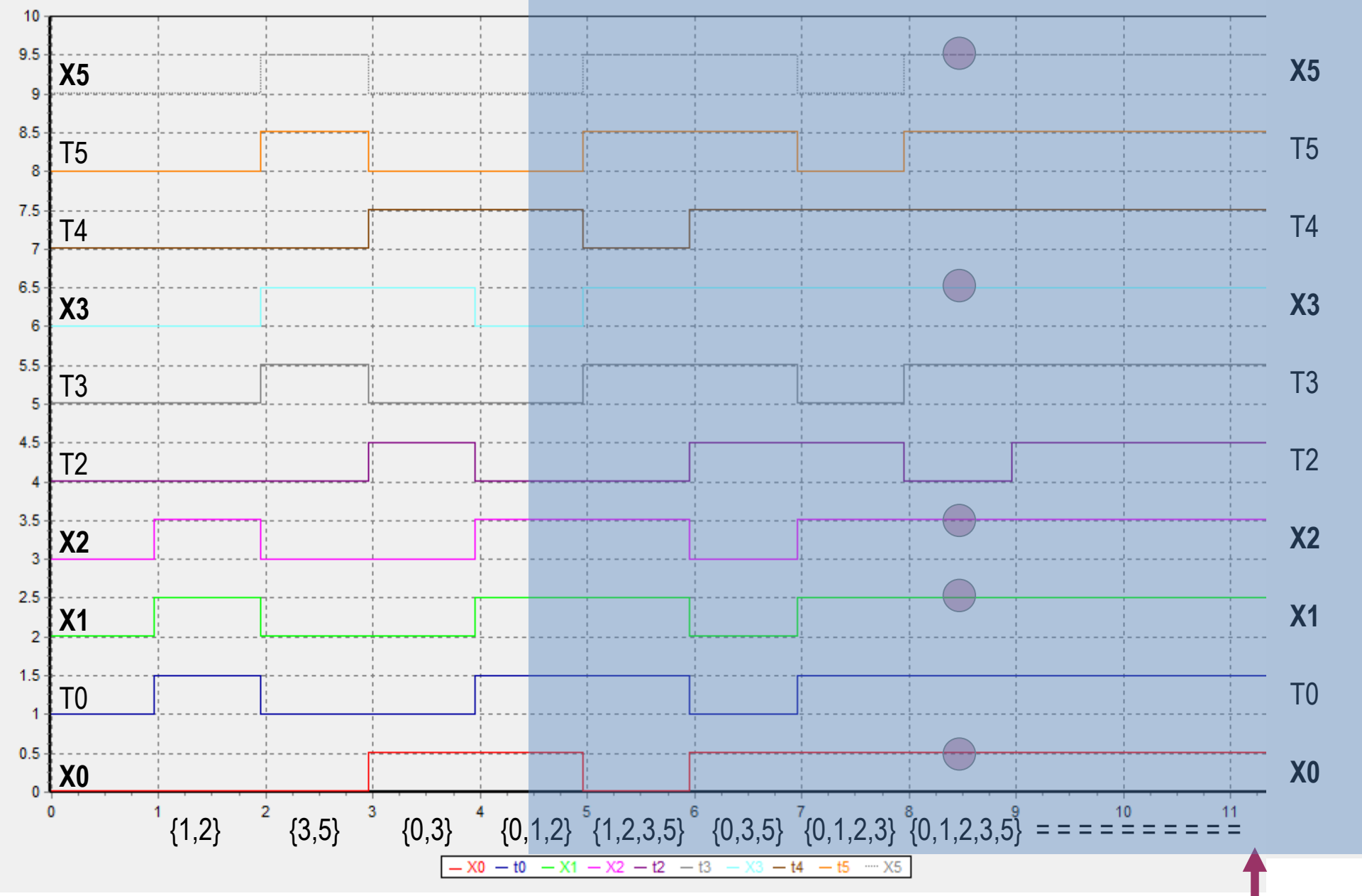
(1 segundo por ciclo) Todas as etapas ligadas...



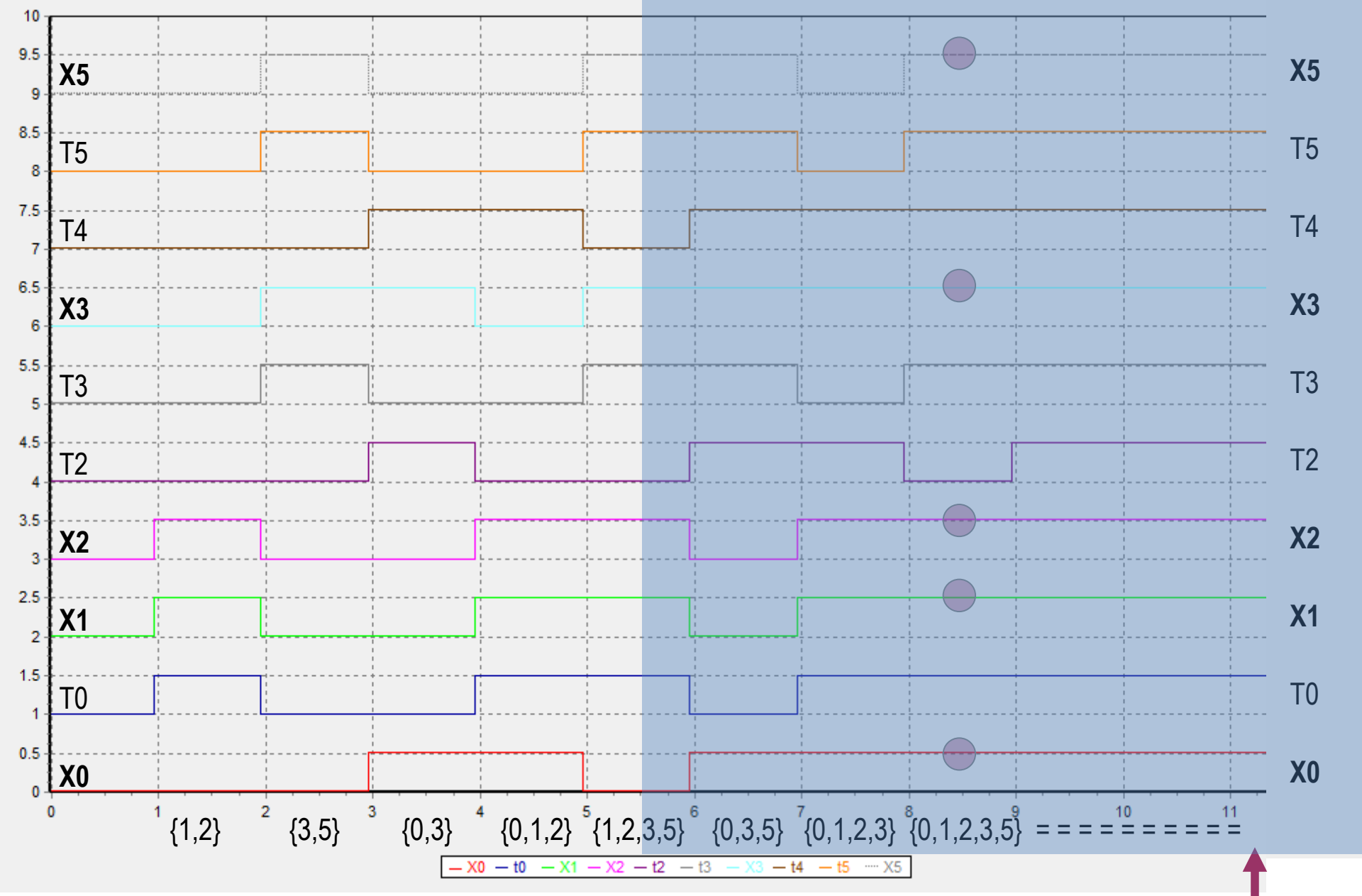
(1 segundo por ciclo) Todas as etapas ligadas...



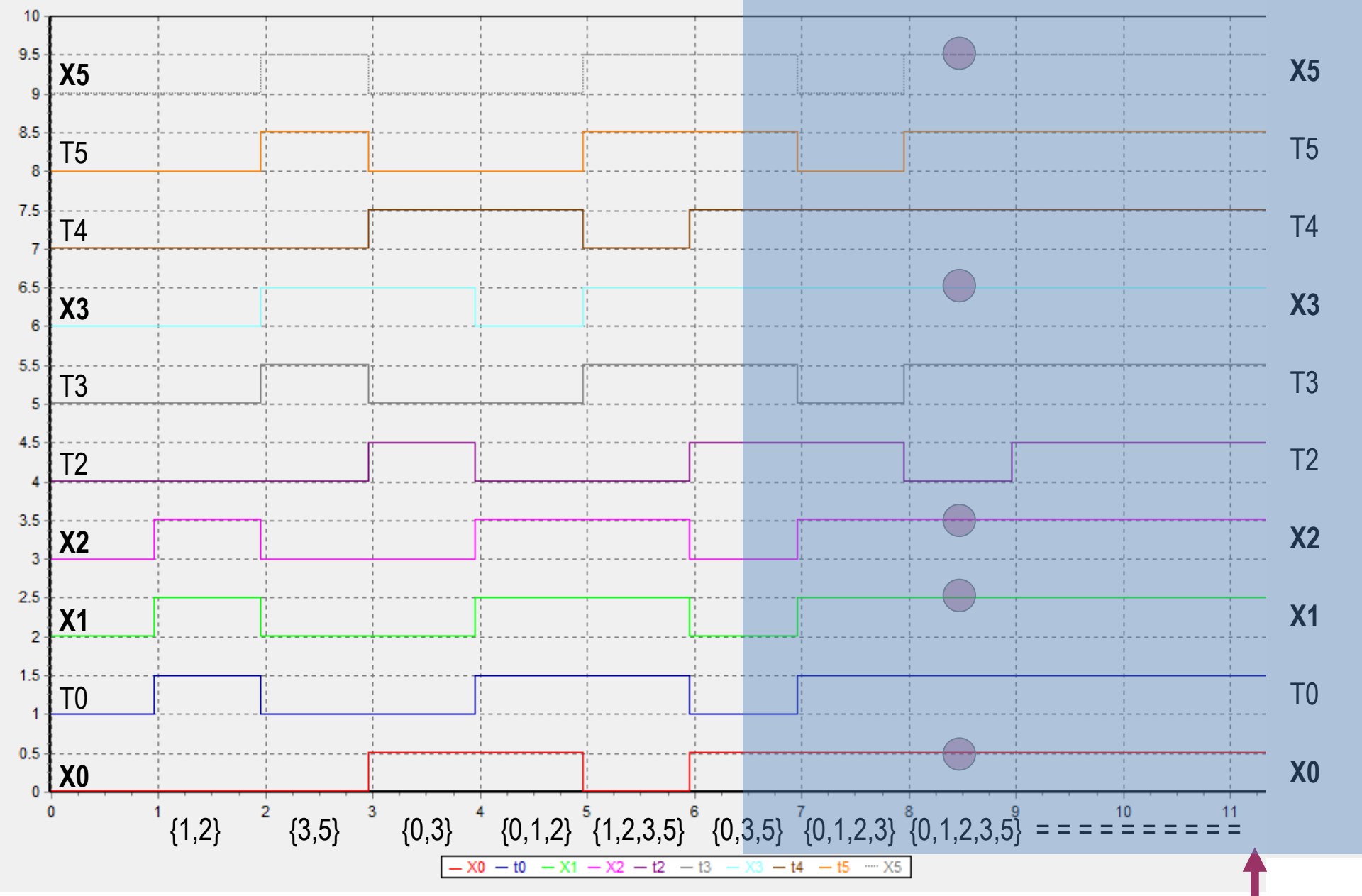
(1 segundo por ciclo) Todas as etapas ligadas...



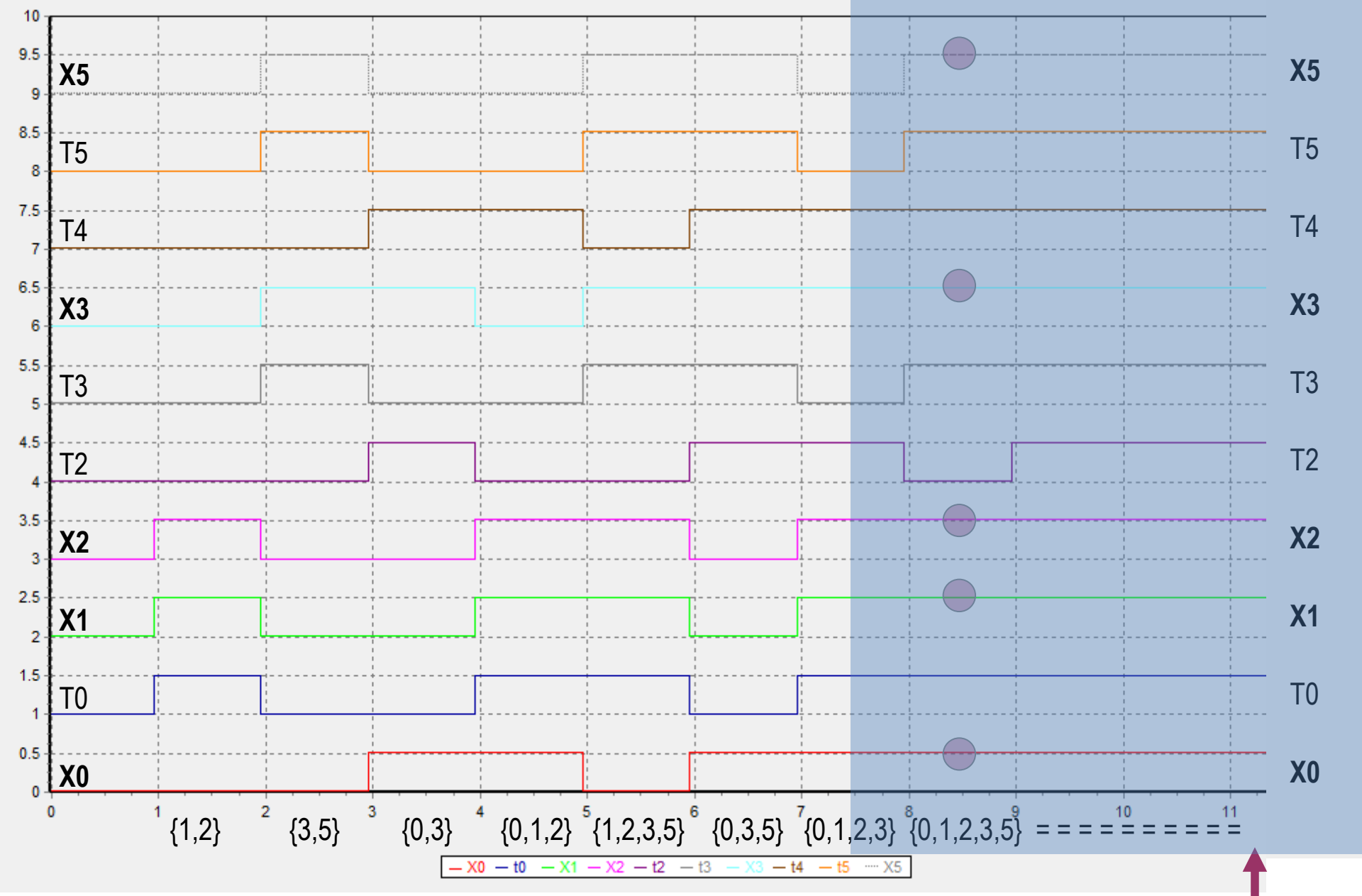
(1 segundo por ciclo) Todas as etapas ligadas...



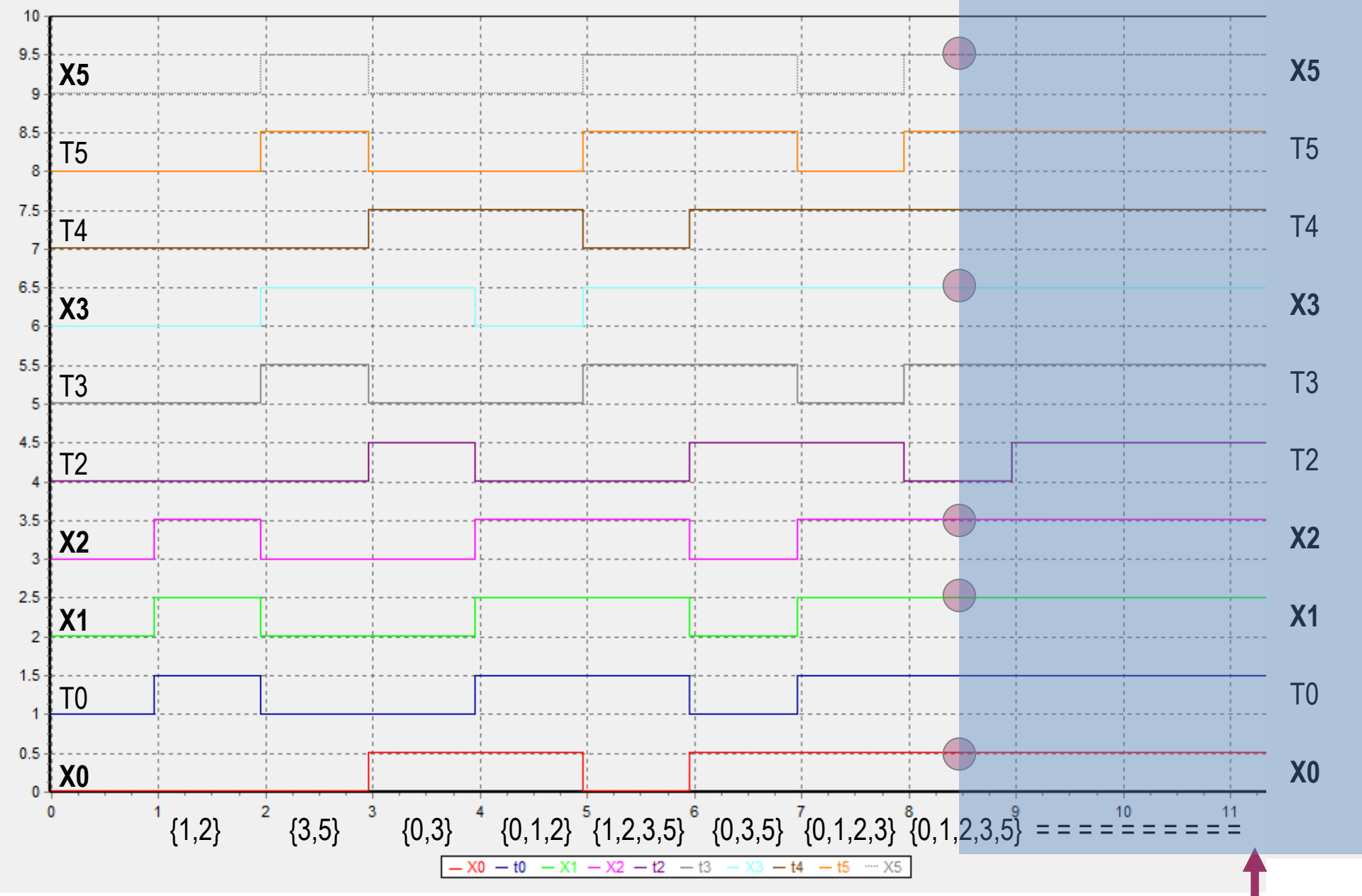
(1 segundo por ciclo) Todas as etapas ligadas...



(1 segundo por ciclo) Todas as etapas ligadas...



(1 segundo por ciclo) Todas as etapas ligadas...



(1 segundo por ciclo) Todas as etapas ligadas...

Limitações FEUPAutomGrafcet

Obs:

O algoritmo apresentado permite alguma hierarquia (entre páginas)

porém não permite … um Grafcet congelar outro ☹ !

=> Transições "congeladas" por página

=> Daí a necessidade dos hook's ("Zones")

FEUPAutom - Control Cycle

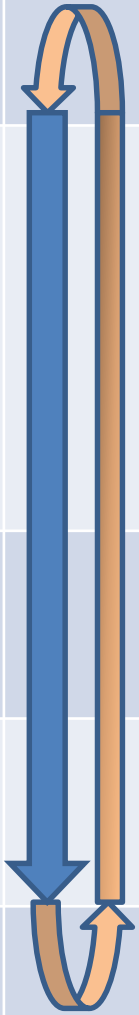
Read from Physical Inputs
to Memory Variables

Execution of Script -
FEUPAutom / FEUPAutomGrafcet
(detailed next)

Write to Physical Outputs
from Memory Variables

Internal Processing

Operating System (Windows/…)



All
Processing
is done
on memory
variables

FEUPAutomGrafcet

Zones

*** Repetir da Página 3 à Página 0 ***

Se (Cold_Boot) => Ativar etapas iniciais

Se (não Cold_Boot) => Calc. Trans. Disparadas (TrD)

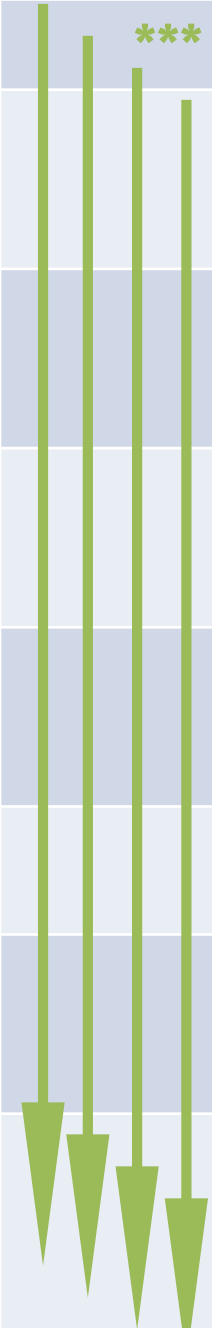
Para as TrDs, desligar etapas a montante

Para as TrDs, ligar etapas a jusante + ini. temporizad.

Se (Página_P3) => Desligar todas saídas

Cada 1/10 de seg. para Etapas Ativas=>
=> Incrementar temporiz. dessa Etapa

Para Etapas Ativas => Executar ação



"Zone1"

"Zone2"

"Zone3"

"Zone4"

"Zone5"

"Zone6"

"Zone7"

"Zone8"

“Compilar”

- Criar variáveis com nomes certos
  - 1 bit / transição (t_i)
  - 1 bit / etapa (x_j)
  - 1 word / etapa (x_j_T)
- Para cada Transição
  (tendo em consideração saltos)
  - Congelamento por página
  - Listar etapas a montante
  - Listar etapas a jusante
- Gerar efetivamente o código ☺

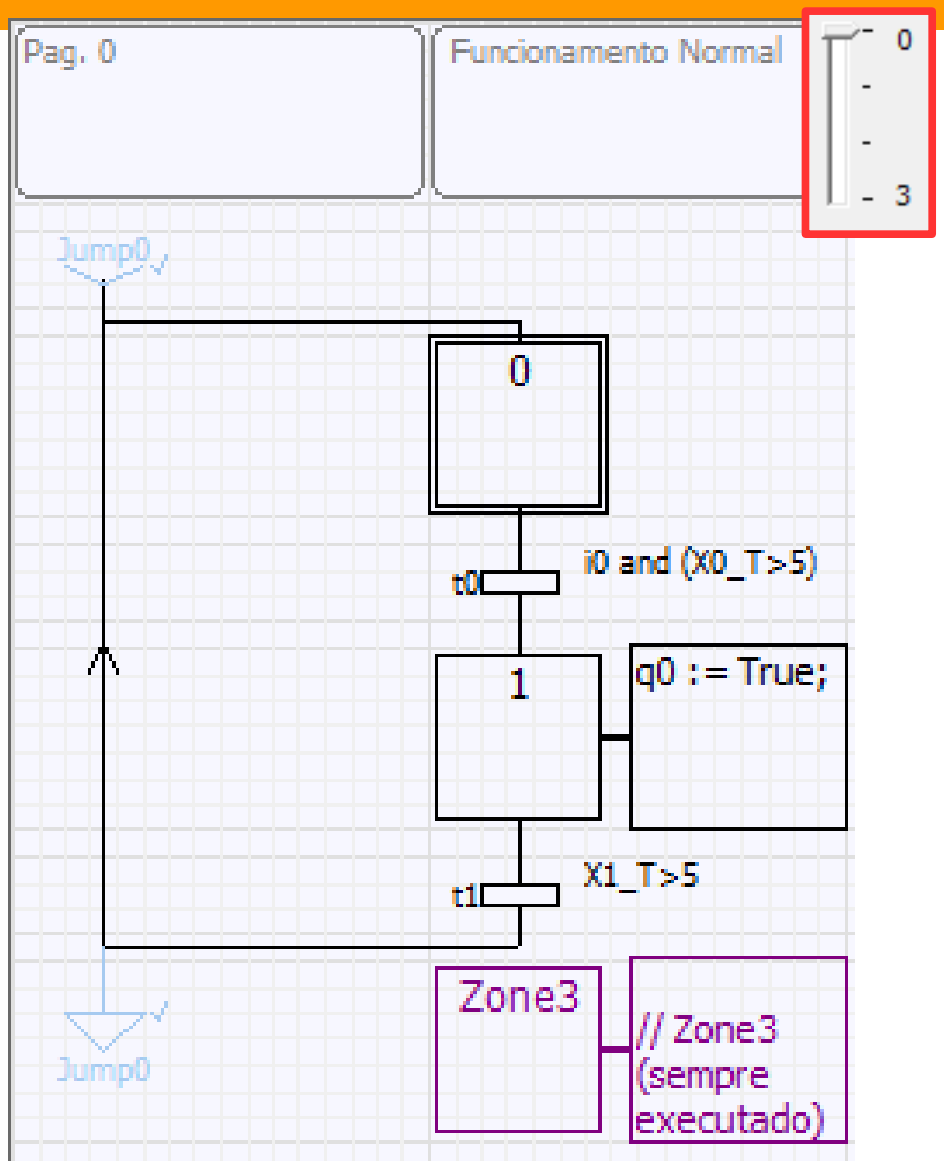
Atenção: é possível utilizar variáveis auxiliares
mas não devem estar em conflito com o Grafcet

Limitações FEUPAutomGrafcet

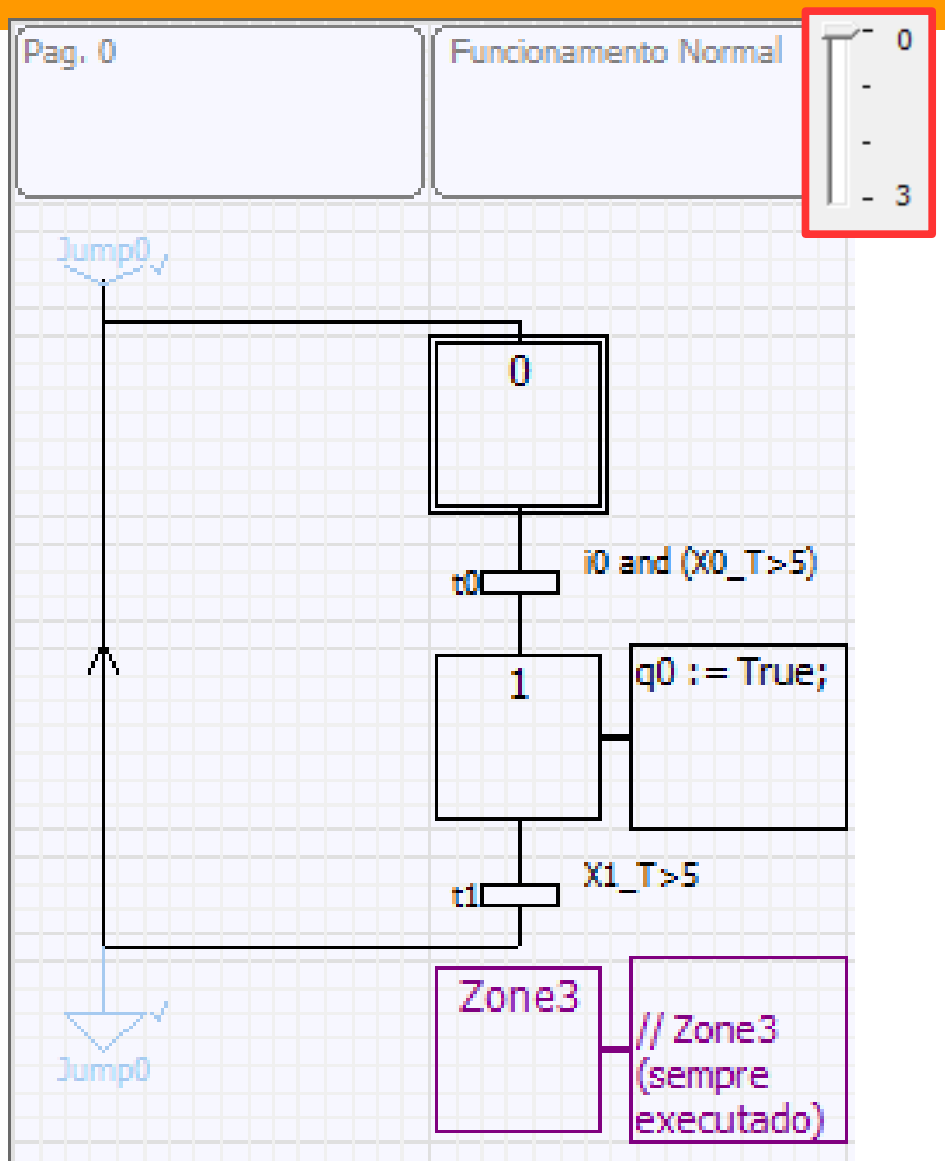
Com estes hooks, é possível “congelar” uma parte do Grafcet com código na “Zone ??? ” ☺

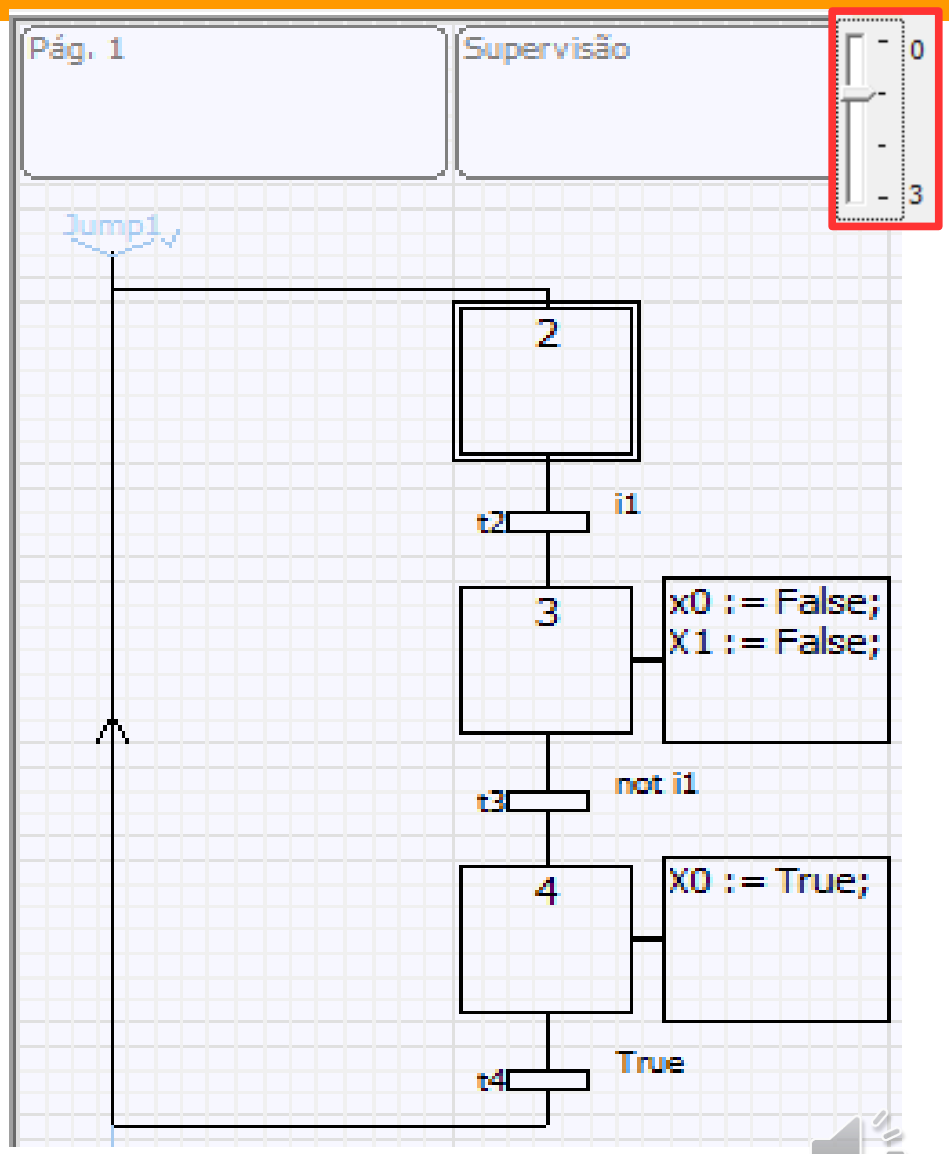
Com estes hooks, é possível escrever verificações de segurança ao código na “Zone ??? ” ☺

Páginas (próximo de hierarquia)

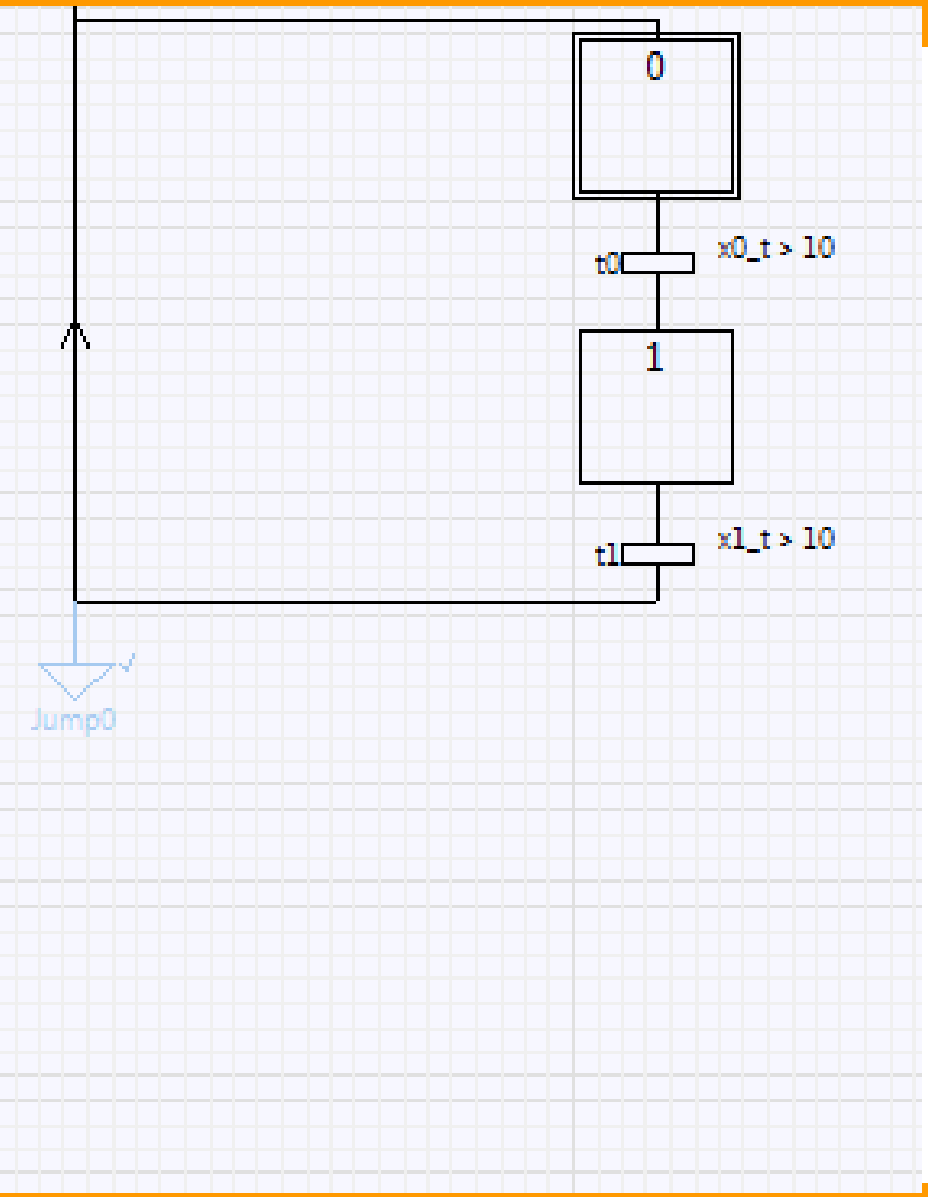


Páginas (próximo de hierarquia)



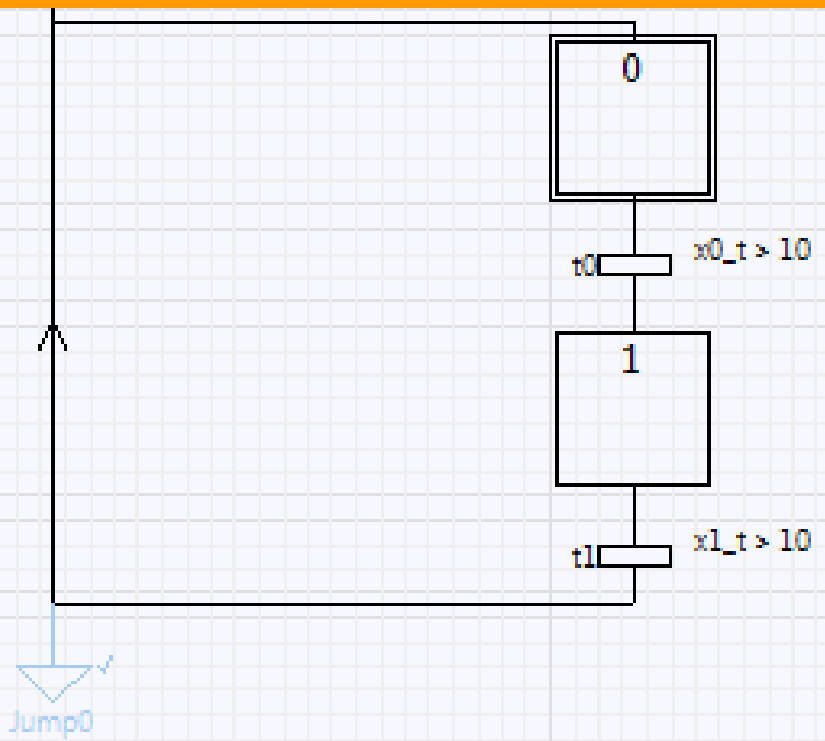


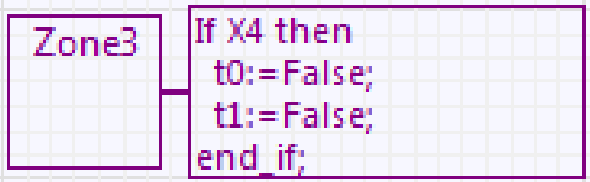
Congelamento

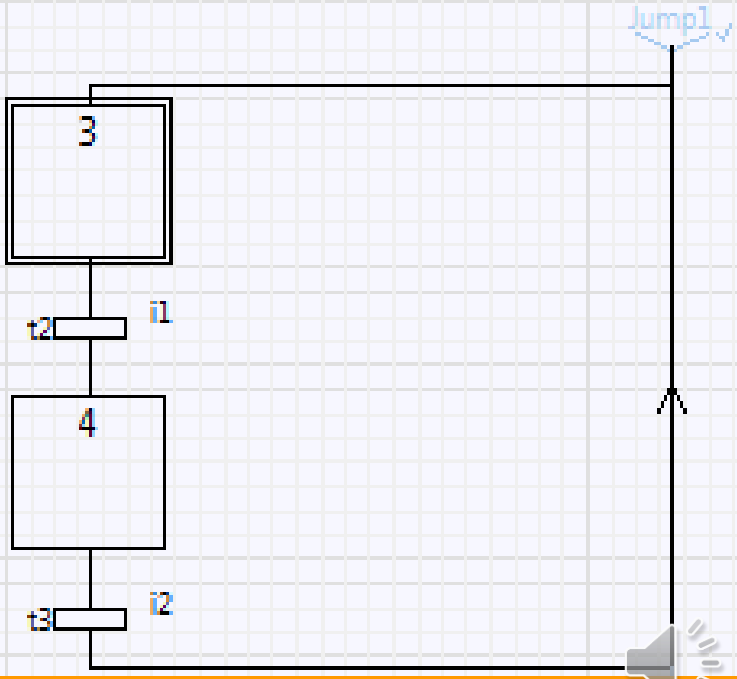


Congelamento

FEUPAutom < versão 5.50





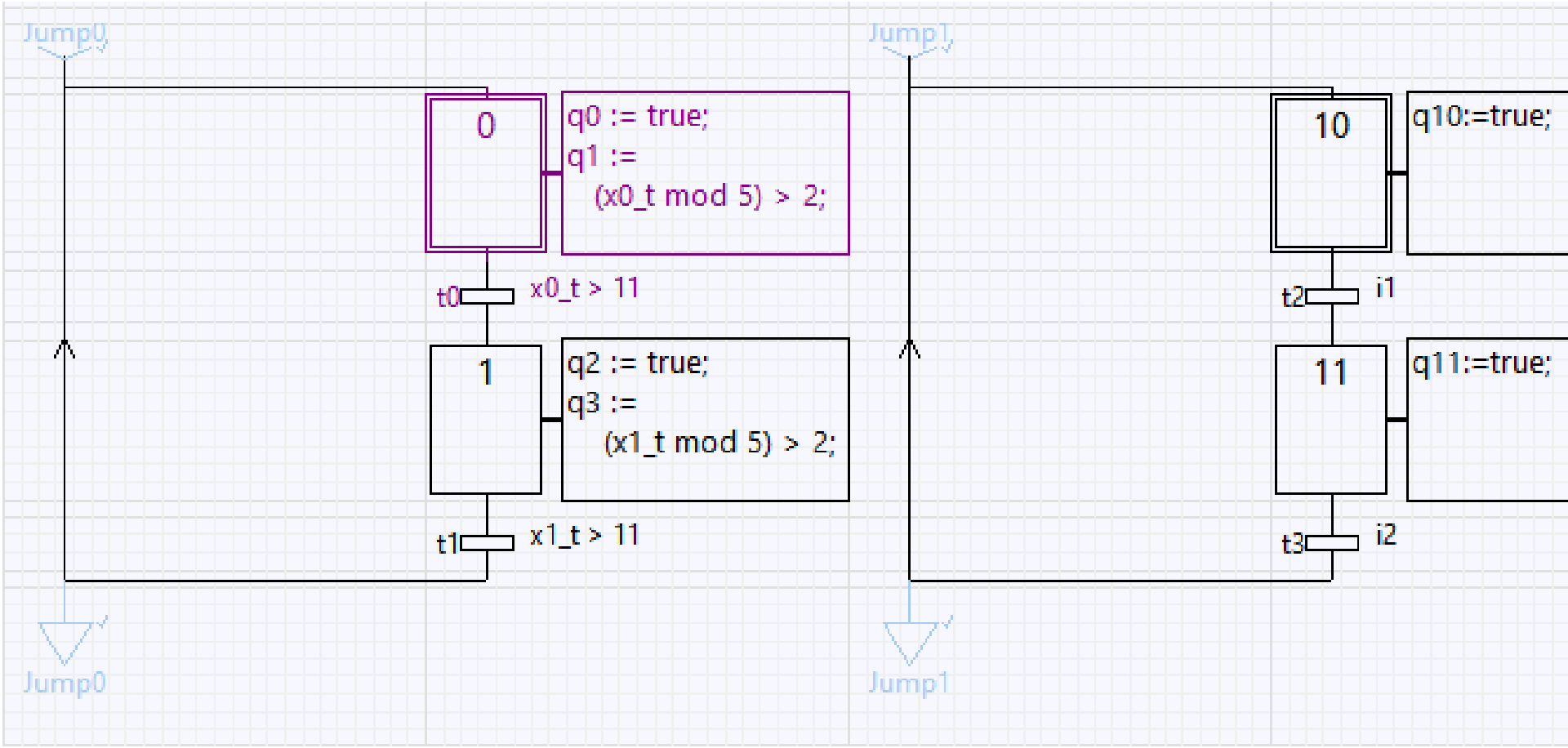


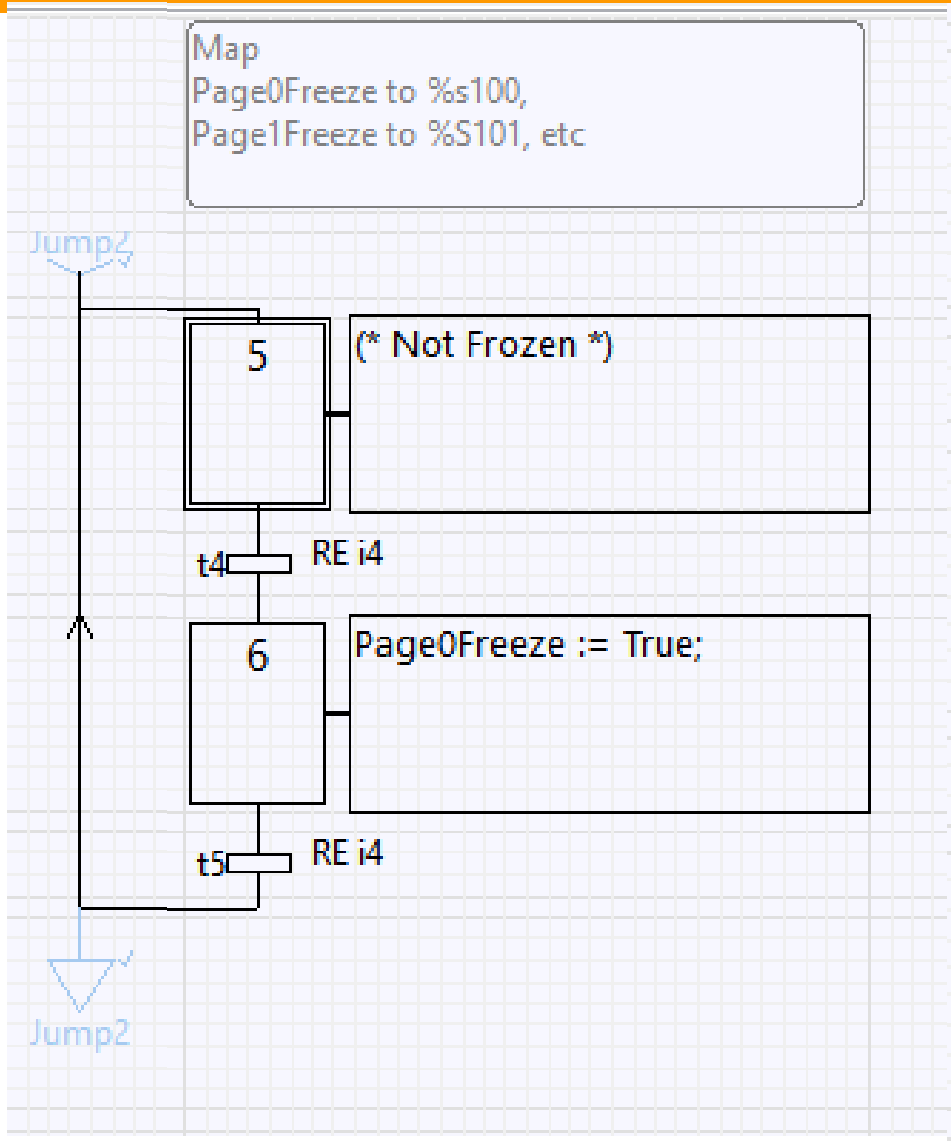
Atenção:
A “Zone3” tem de estar na página das transições a alterar

Freeze, PageStart/Page Clear

FEUPAutom v.5.50+

Página 0





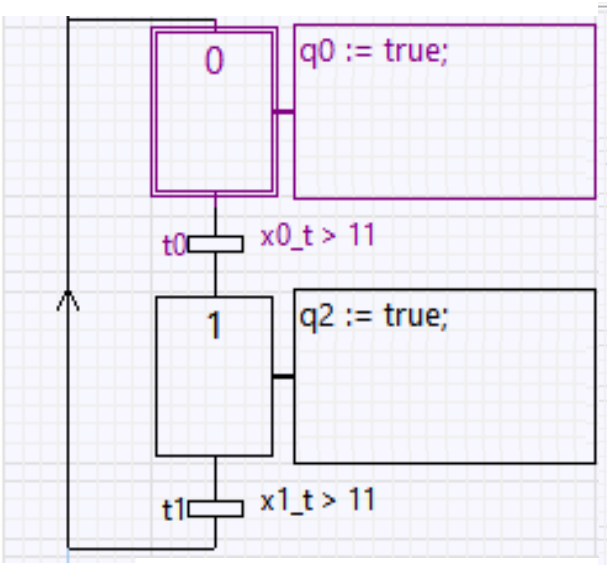

```
100
if RE i6 then
  PageStart(0);
  (*Set Ini Steps + allow evol.*)
end_if;
```

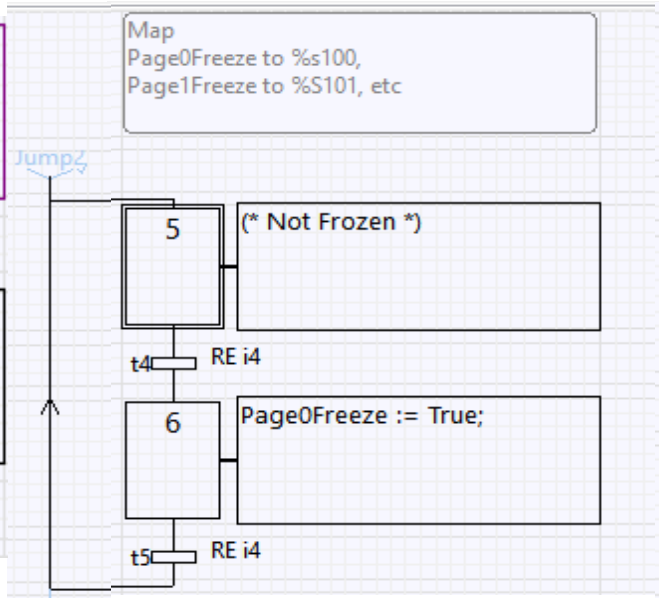
```
101
if i7 then
  PageClear(0);
end_if;
```

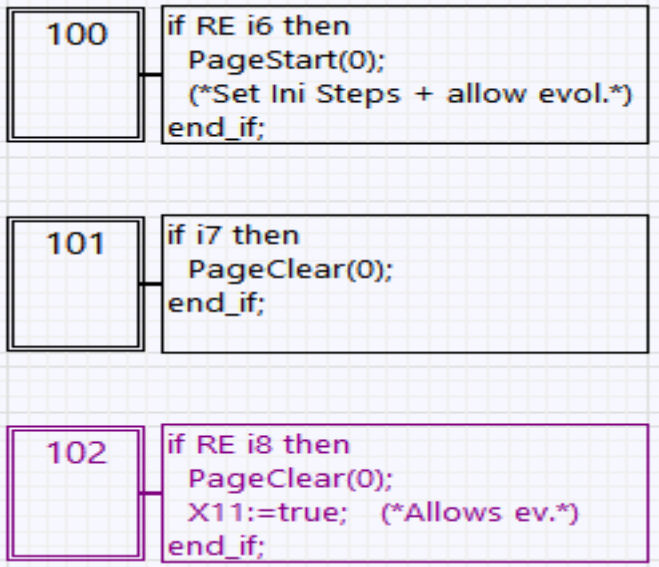
```
102
if RE i8 then
  PageClear(0);
  X11:=true;   (*Allows ev.*)
end_if;
```

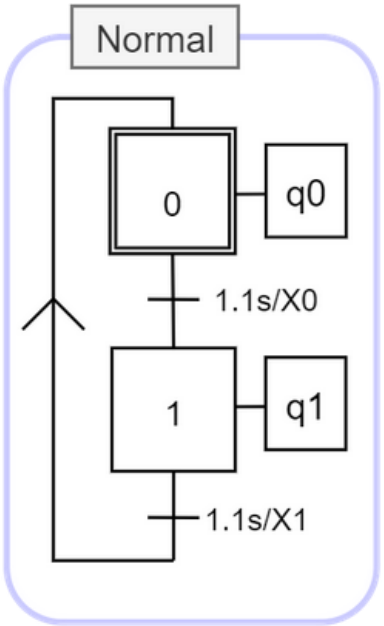
Freeze, PageStart/Page Clear

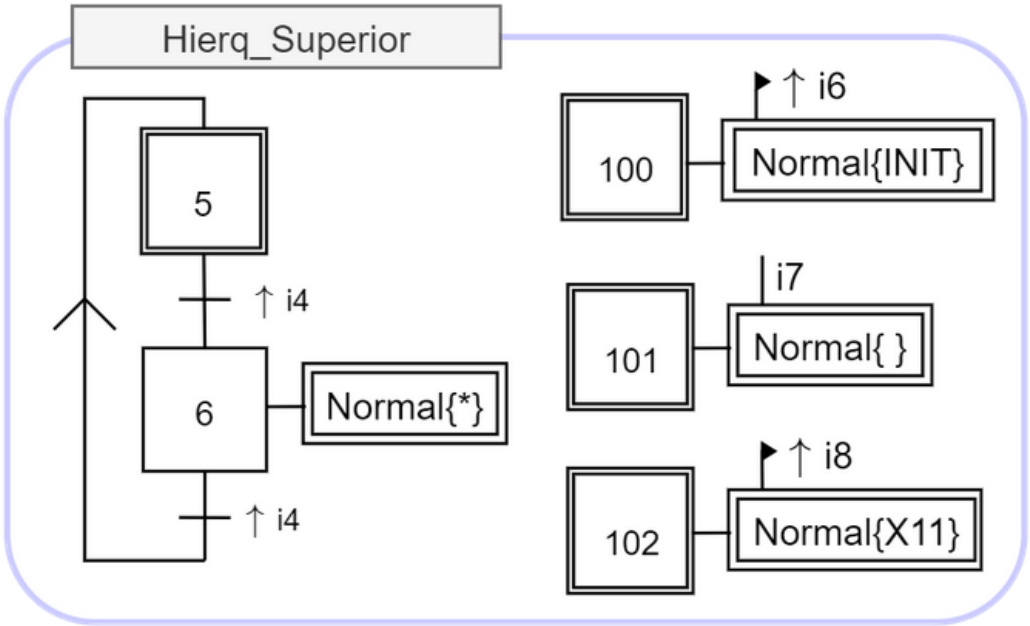
FEUPAutom v.5.50+











FEUPAutom

- ☹ Geração de código “OneWay”
  - ☹ Alterar código do lado gráfico
  - ☹ As erros têm de ser consultados na janela de ST;
  - ☹ Compilar Grafcet faz desaparecer alterações ao ST)

- ☹ Não tem Grafcet hierárquico puro (não tem macro ações)
- ☹ Não tem bits de sistema macro (tal como o Gr7 do PL7)
- ☹ Não tem timers universais do Grafcet (nem TOff)
- ☹ Não tem ações memorizadas
- ☹ Não tem ações impulsionais

FEUPAutom

☺ Estilo próximo da Norma do Grafcet

☺ Tem Páginas de Grafcet (hierarquia limitada)
(permite por exemplo des+ligar etapas)

☺ Tem Hooks (permite por exemplo congelar o Grafcet)

☺ Tem um tempo (décimos segundo) por etapa

☺ Pode gerar código C (entrada ainda em ST)

☺ Há SystemBit para Congelar por página (%s100, %s101, etc)

☺ Há função para arrancar por página (PageStart(…))

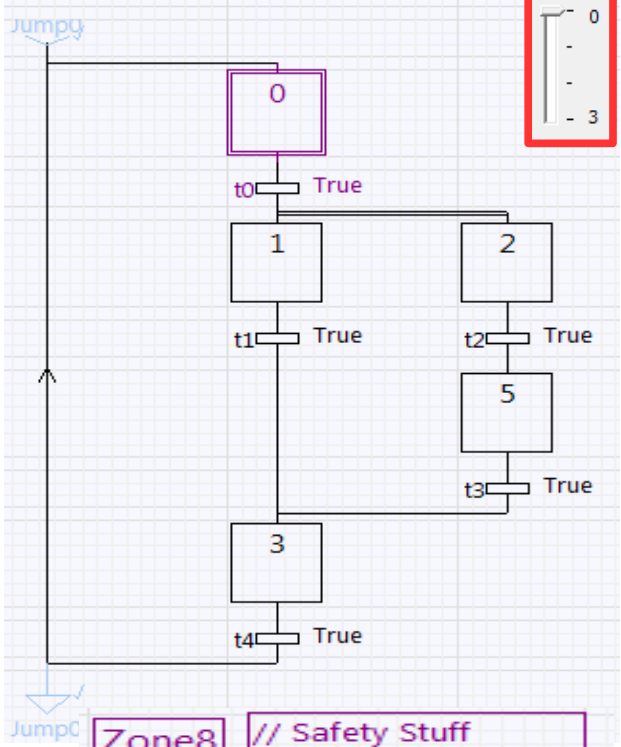
☺ Há função para limpar por página (PageClear(…))

Caso de estudo

Compilar…

Executar…

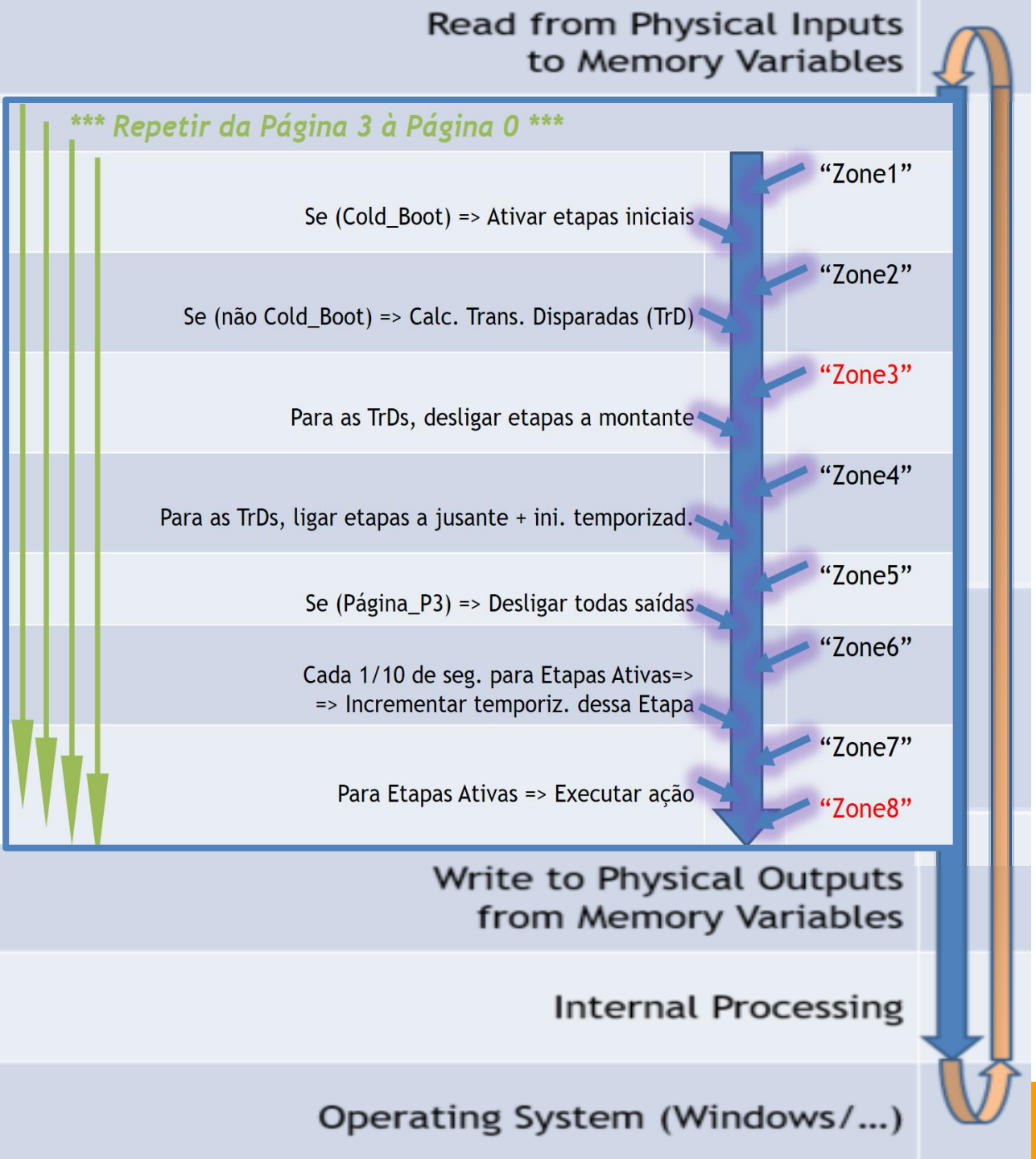
%Name	UserName	%Name	UserName
%M0	X0	%MW0	X0_t
%M1	t0	%MW1	MW1
%M2	X1	%MW2	X1_t
%M3	t1	%MW3	MW3
%M4	M4	%MW4	MW4
%M5	M5	%MW5	MW5
%M6	Zone3	%MW6	Zone3_t
%M7	X9	%MW7	X9_t
%M8	t2	%MW8	MW8
%M9	X4	%MW9	X4_t
%M10	t3	%MW10	MW10
%M11	M11	%MW11	MW11
%M12	M12	%MW12	MW12
%M13	X5	%MW13	X5_t
%M14	t4	%MW14	MW14
%M15	X6	%MW15	X6_t
%M16	t5	%MW16	MW16
%M17	M17	%MW17	MW17
%M18	M18	%MW18	MW18
%M19	Zone8	%MW19	Zone8_t
%M20	t6	%MW20	MW20
%M21	X99	%MW21	X99_t
%M22	M22	%MW22	MW22
%M23	M23	%MW23	MW23
%M24	M24	%MW24	MW24
%M25	M25	%MW25	MW25




```
Zone8
// Safety Stuff
if q0 and q1 then
  q0:=false;
end_if;
```

```
Zone3
If X4 then
  t0:=False;
  t1:=False;
  t2:=False;
  t3:=False;
  t4:=False;
end_if;
```

```
99
// Do Stuff
q0:=i0;
```


Read from Physical Inputs
to Memory Variables
*** Repetir da Página 3 à Página 0 ***
"Zone1"
Se (Cold_Boot) => Ativar etapas iniciais
"Zone2"
Se (não Cold_Boot) => Calc. Trans. Disparadas (TrD)
"Zone3"
Para as TrDs, desligar etapas a montante
"Zone4"
Para as TrDs, ligar etapas a jusante + ini. temporizad.
"Zone5"
Se (Página_P3) => Desligar todas saídas
"Zone6"
Cada 1/10 de seg. para Etapas Ativas=>
=> Incrementar temporiz. dessa Etapa
"Zone7"
Para Etapas Ativas => Executar ação
"Zone8"
Write to Physical Outputs
from Memory Variables
Internal Processing
Operating System (Windows/...)

Read from Physical Inputs
to Memory Variables

*** Repetir da Página 3 à Página 0 ***

"Zone1"

Se (Cold_Boot) => Ativar etapas iniciais

"Zone2"

Se (não Cold_Boot) => Calc. Trans. Disparadas (TrD)

"Zone3"

Para as TrDs, desligar etapas a montante

"Zone4"

Para as TrDs, ligar etapas a jusante + ini. temporizad.

"Zone5"

Se (Página_P3) => Desligar todas saídas

"Zone6"

Cada 1/10 de seg. para Etapas Ativas=>
=> Incrementar temporiz. dessa Etapa

"Zone7"

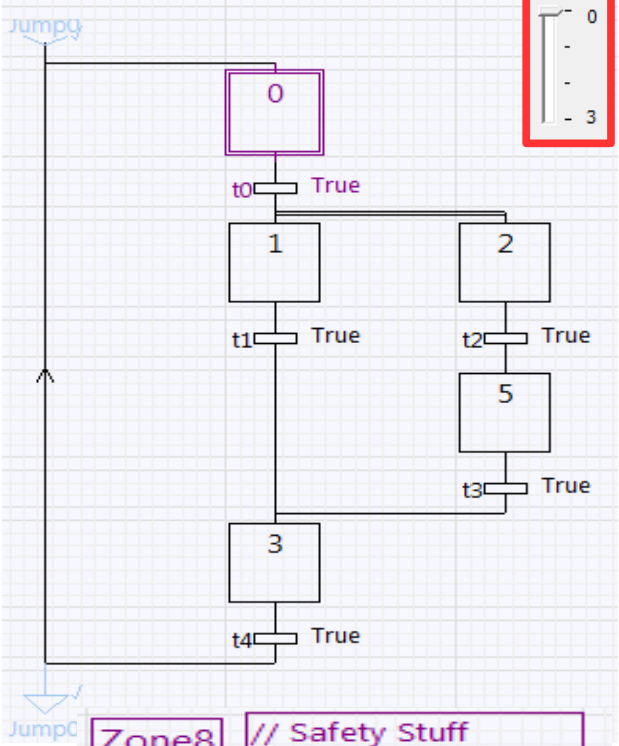
Para Etapas Ativas => Executar ação

"Zone8"

Write to Physical Outputs
from Memory Variables

Internal Processing

Operating System (Windows/...)

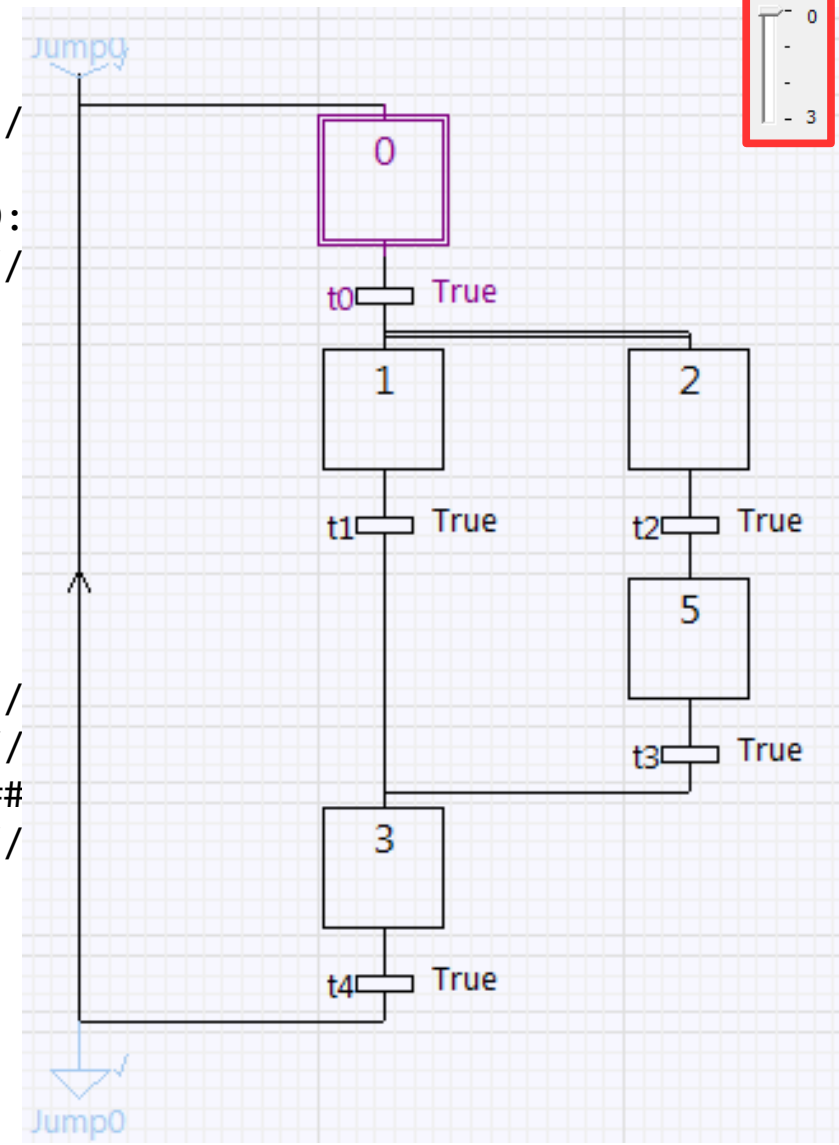



```
Zone8
// Safety Stuff
if q0 and q1 then
  q0:=false;
end_if;
```

```
Zone3
If X4 then
  t0:=False;
  t1:=False;
  t2:=False;
  t3:=False;
  t4:=False;
end_if;
```

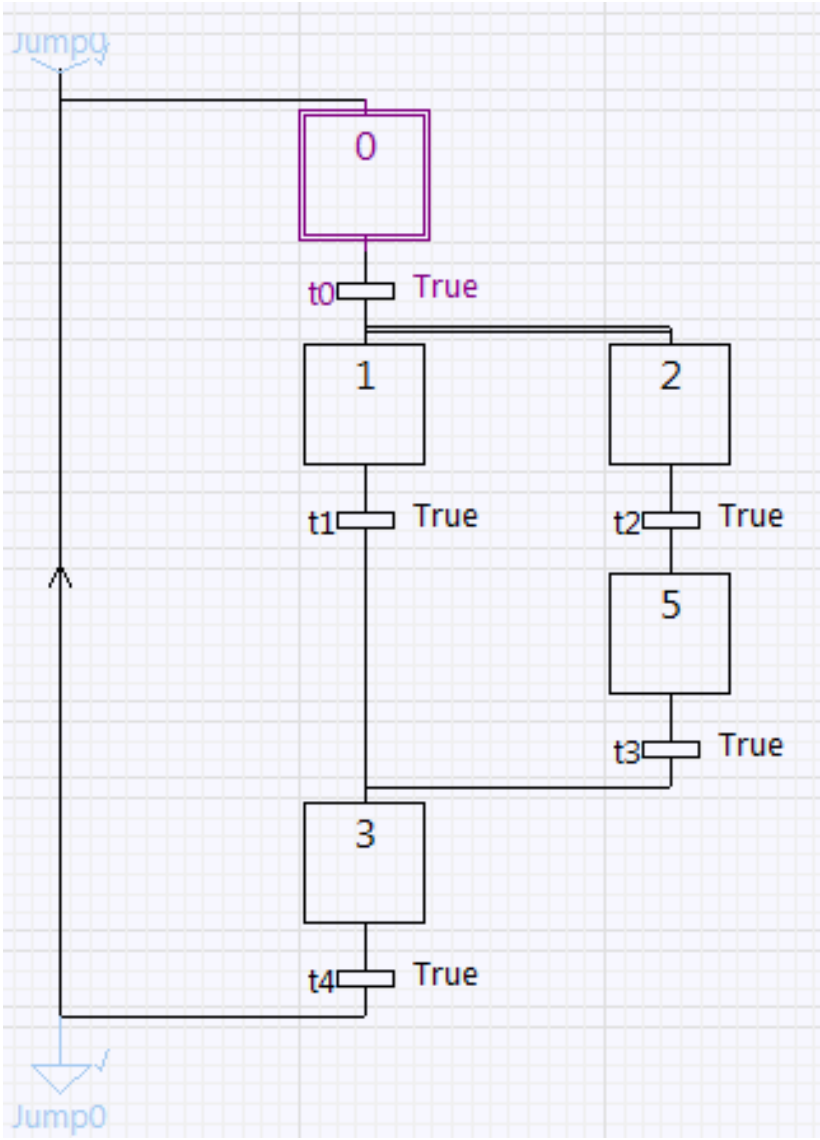
```
99
// Do Stuff
q0:=i0;
```

```
////////////////////////////////////////////////
// FEUPAutom - 5.9.0.98 -
// Code Automatically Generated:2019-04-04 16:09:
////////////////////////////////////////////////



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

////////////////////////////////////////////////
////////////// If boot => Set Initial Steps //////
//######################## Page 3 ################
////////////////////////////////////////////////

  If (sw0=0) Then
  End_If;
```



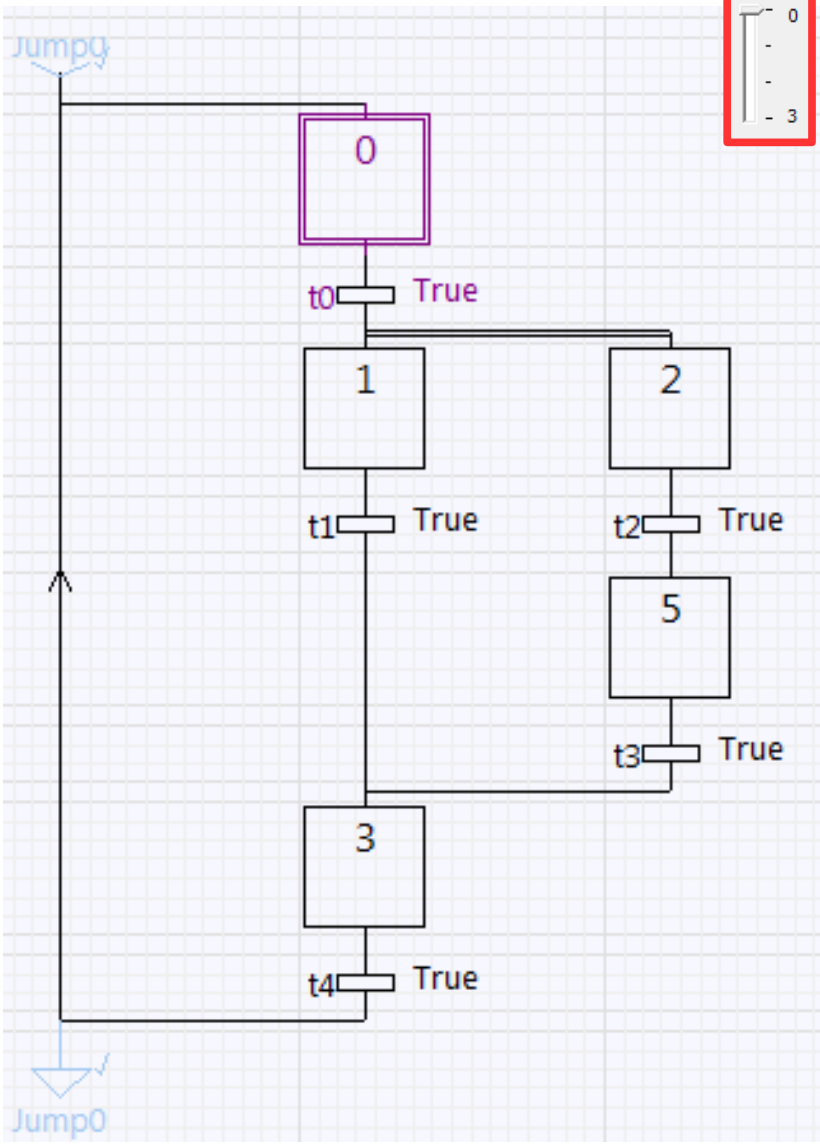
```
Q0:=False;
Q1:=False;
Q2:=False;
Q3:=False;
Q4:=False;
...
```

```
%s100 := False;
%s101 := False;
%s102 := False;
```

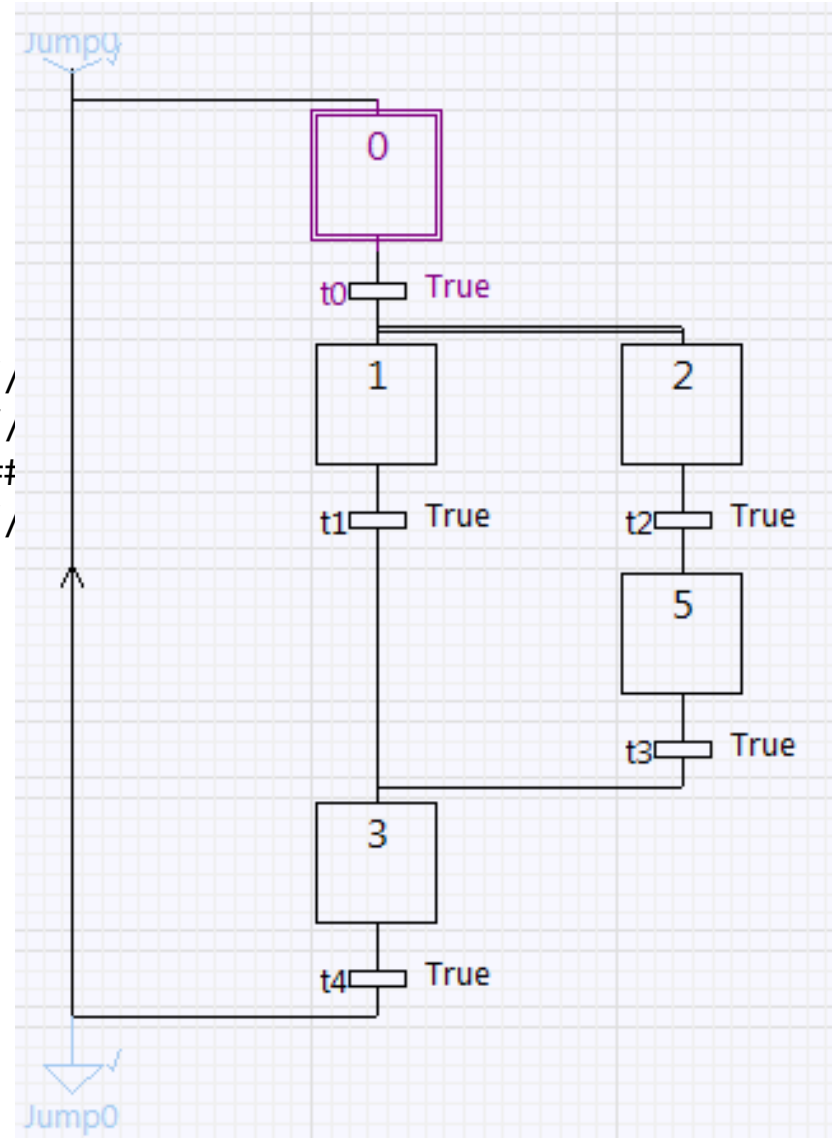


```
//########################################//
//################ Page 2 ###############//
//########################################//

...
```

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

...
```



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


////////////////////////////////////////////////
//////// If boot => Set Initial Steps ////////
//#################### Page 0 ##################
////////////////////////////////////////////////

  If (sw0=0) Then
    // ObjIdx=0 => INI_Step "X0"
    X0 := True;
  End_If;
```

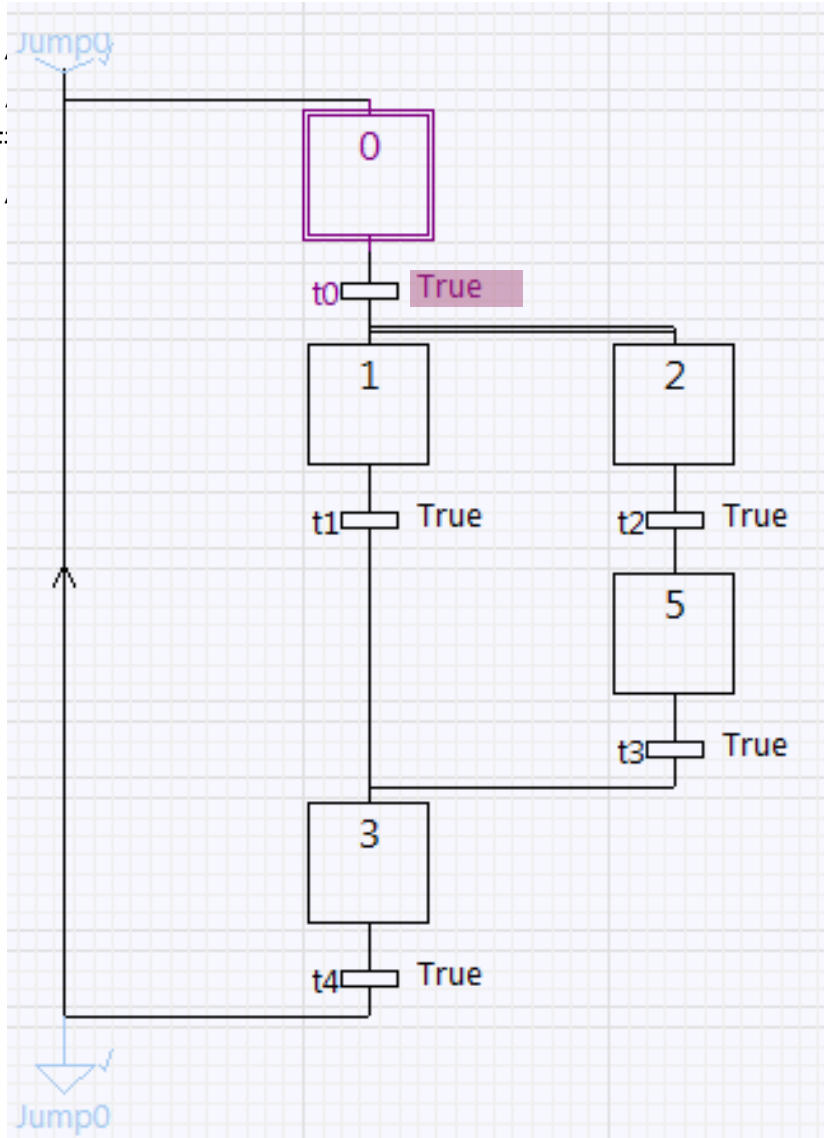


Obs:

Id=ObjIdx=0 =>

%M0 é X0 e %MW0 é X0_t

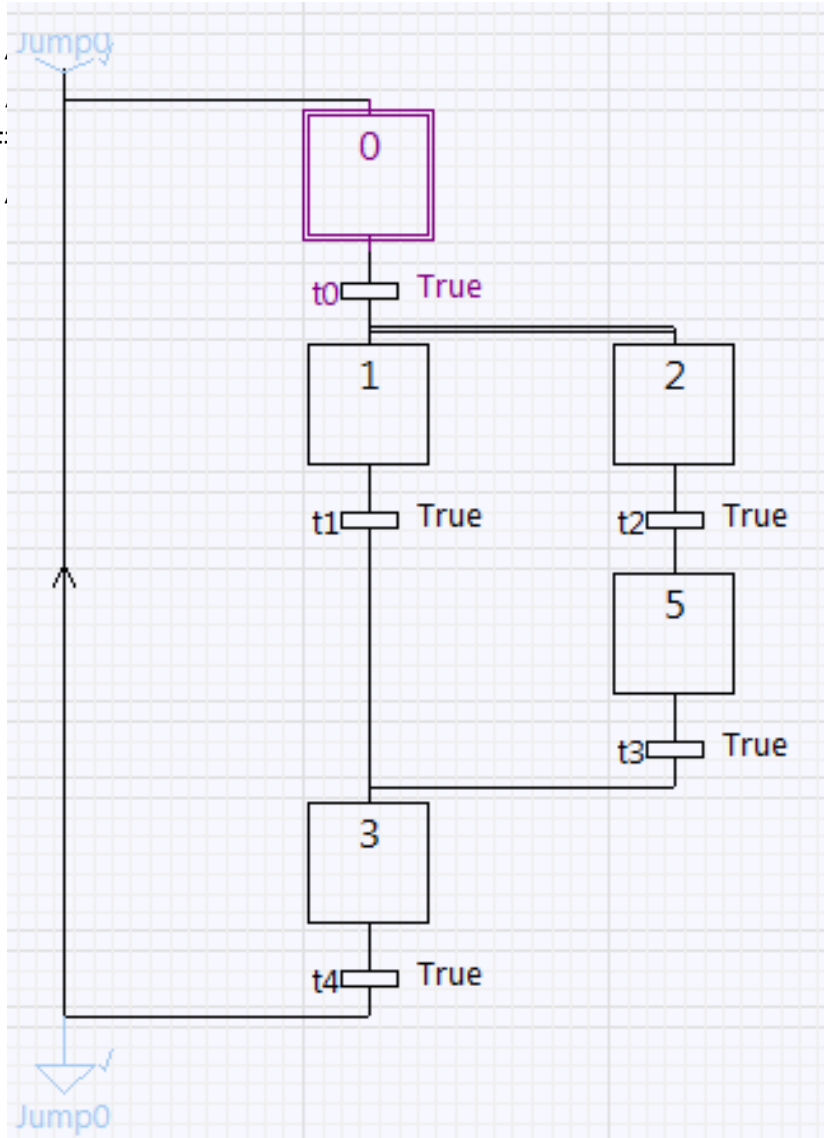
```
////////////////////////////////////////////////
///////////////// Calc Fired Transitions ///////
//####################### Page 0 ###############
////////////////////////////////////////////////

// ObjIdx=1 => Transition "t0"
  // Steps Above: id=0 => X0 ;
  t0 :=  X0 AND  (True);

// ObjIdx=4 => Transition "t1"
  // Steps Above: id=2 => X1 ;
  t1 :=  X1 AND  (True);

// ObjIdx=5 => Transition "t2"
  // Steps Above: id=3 => X2 ;
  t2 :=  X2 AND  (True);

// ObjIdx=7 => Transition "t3"
  // Steps Above: id=6 => X5 ;
  t3 :=  X5 AND  (True);

// ObjIdx=9 => Transition "t4"
  // Steps Above: id=8 => X3 ;
  t4 :=  X3 AND  (True);
```



```
////////////////////////////////////////////////
///////////////// Calc Fired Transitions ///////
//####################### Page 0 ###############
////////////////////////////////////////////////
```

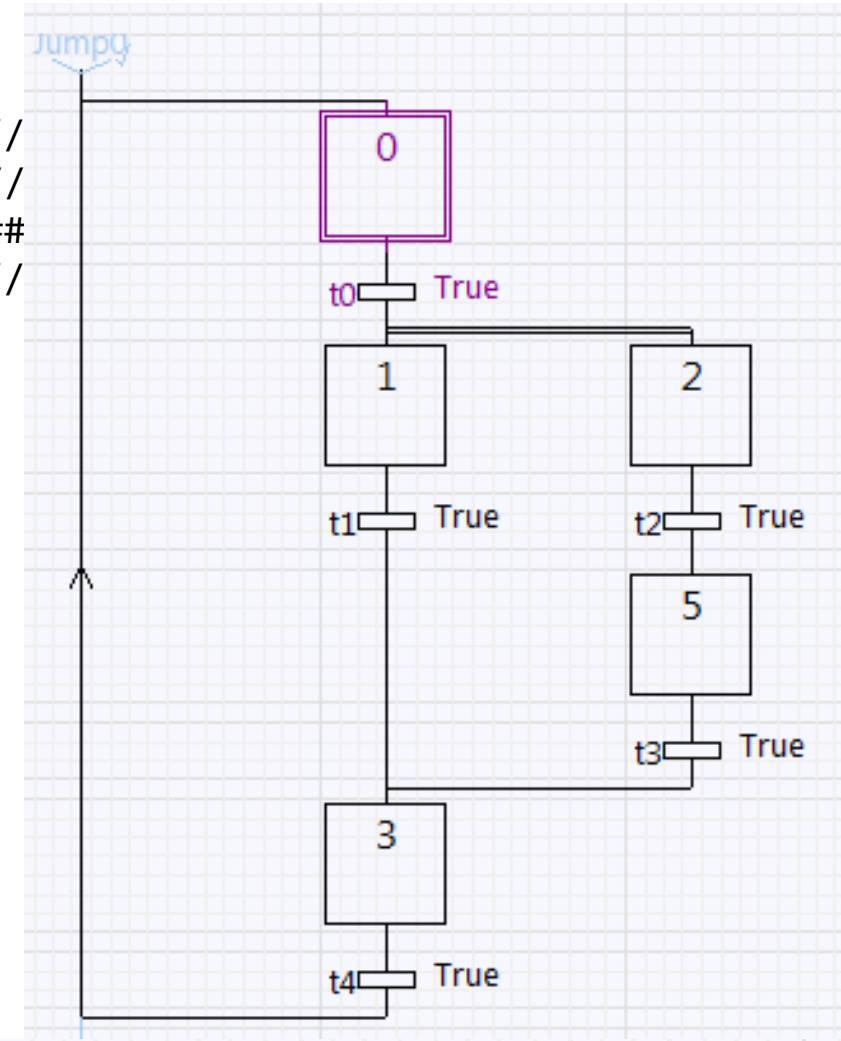
// FEUPAutom v.5.50+

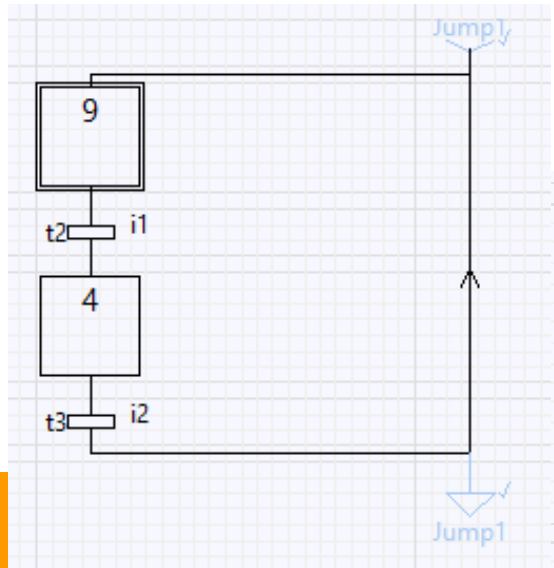
```
// ObjIdx=1 => Transition "t0"
  // Steps Above: id=0 => X0 ;
  t0 := (NOT %s100) AND ( X0 AND  (True) );

// ObjIdx=4 => Transition "t1"
  // Steps Above: id=2 => X1 ;
  t1 := (NOT %s100) AND ( X1 AND  (True) );

// ObjIdx=5 => Transition "t2"
  // Steps Above: id=3 => X2 ;
  t2 := (NOT %s100) AND ( X2 AND  (True) );

// ObjIdx=7 => Transition "t3"
  // Steps Above: id=6 => X5 ;
  t3 := (NOT %s100) AND ( X5 AND  (True) );

// ObjIdx=9 => Transition "t4"
  // Steps Above: id=8 => X3 ;
  t4 := (NOT %s100) AND ( X3 AND  (True) );
```

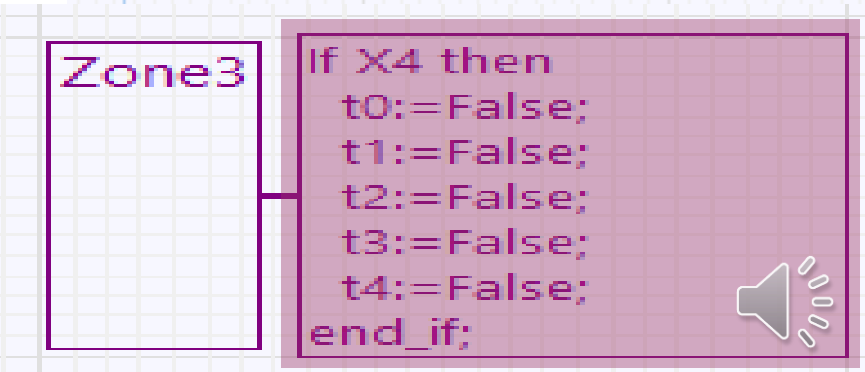


```
////////////////////////////////////////////////
/////////////////////////// Zone3 ///////////////
//######################## Page 0 ###############
////////////////////////////////////////////////
```

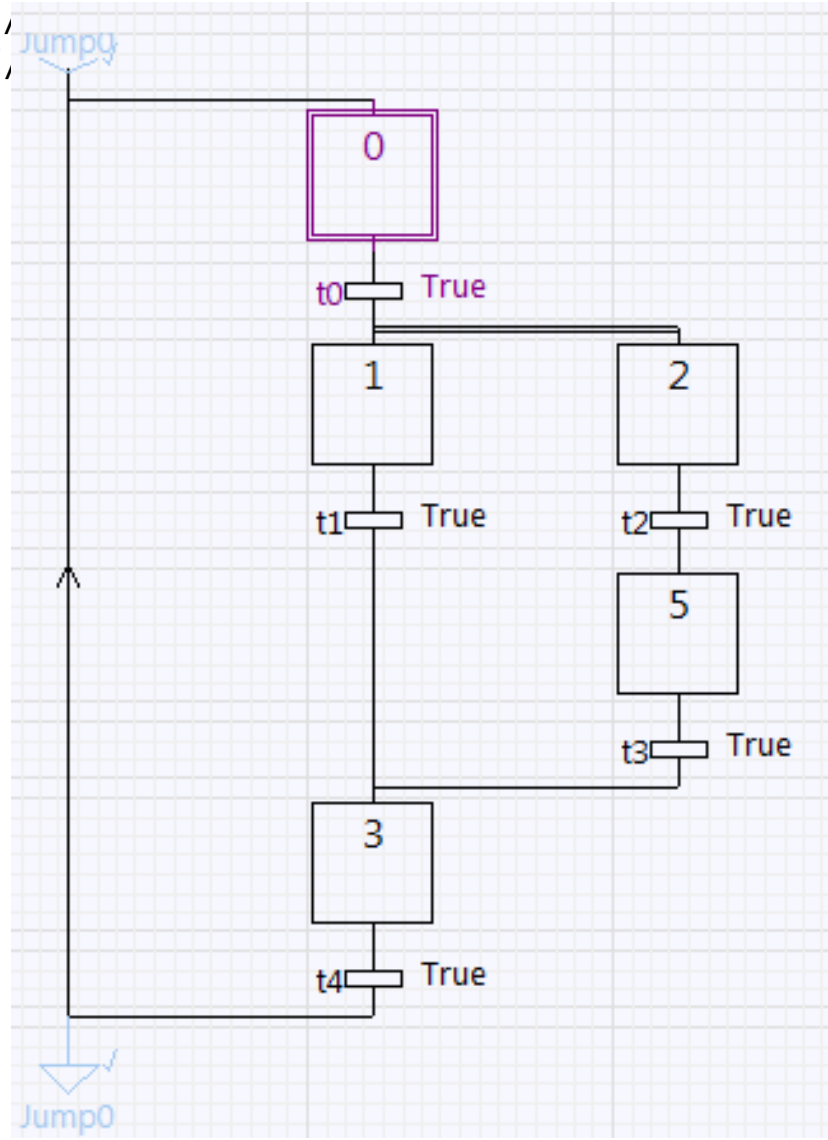
```
If X4 then
   t0:=False;
   t1:=False;
   t2:=False;
   t3:=False;
   t4:=False;
 end_if;
```



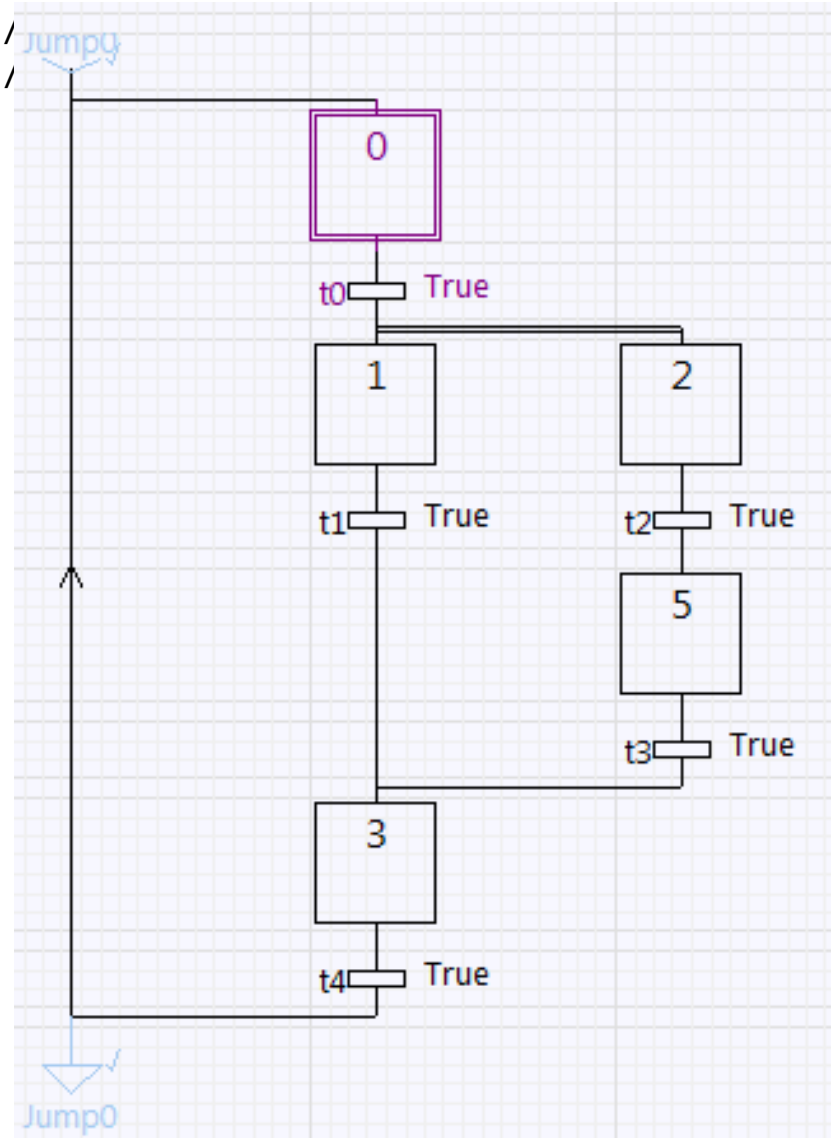




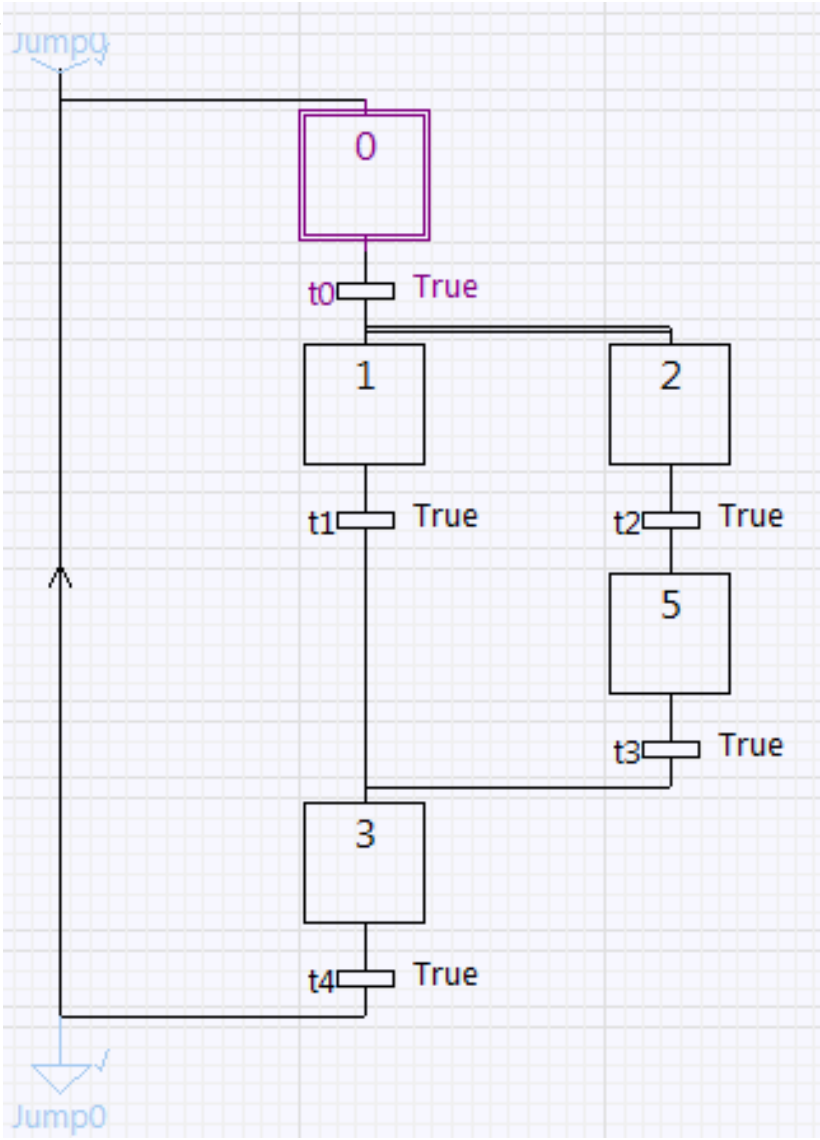
```
////////////////////////////////////////////////////
///////////////// ReSet Steps Above fired Tr //////

// ObjIdx=1 => Transition "t0"
  // Steps Above: id=0 => X0 ;
  If (t0) Then
     X0:=False;
  End_If;
// ObjIdx=4 => Transition "t1"
  // Steps Above: id=2 => X1 ;
  If (t1) Then
     X1:=False;
  End_If;
// ObjIdx=5 => Transition "t2"
  // Steps Above: id=3 => X2 ;
  If (t2) Then
     X2:=False;
  End_If;
// ObjIdx=7 => Transition "t3"
  // Steps Above: id=6 => X5 ;
  If (t3) Then
     X5:=False;
  End_If;
// ObjIdx=9 => Transition "t4"
  // Steps Above: id=8 => X3 ;
If (t4) Then
     X3:=False;
  End_If;
```

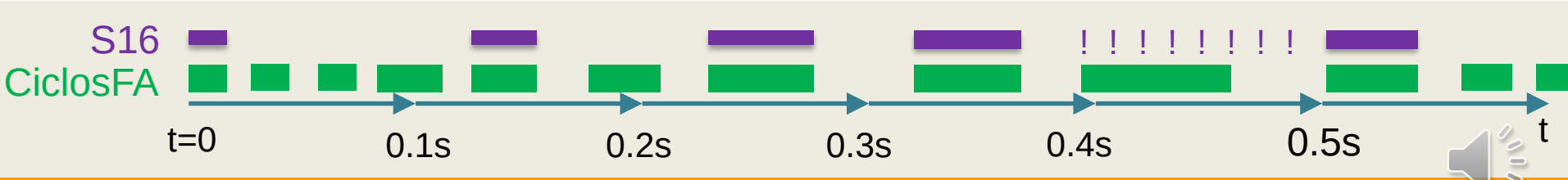


```
////////////////////////////////////////////////////
///////////////// Set Steps below fired Tr /////////

// ObjIdx=1 => Transition "t0"
  // Steps Below: id=2 => X1 ;id=3 => X2 ;
  If (t0) Then
    X1 := True;  X2 := True;
    X1_T := 0; X2_T := 0;
  End_If;
// ObjIdx=4 => Transition "t1"
  // Steps Below: id=8 => X3 ;
  If (t1) Then
    X3 := True;     X3_T := 0;
  End_If;
// ObjIdx=5 => Transition "t2"
  // Steps Below: id=6 => X5 ;
  If (t2) Then
    X5 := True;     X5_T := 0;
  End_If;
// ObjIdx=7 => Transition "t3"
  // Steps Below: id=8 => X3 ;
  If (t3) Then
    X3 := True;     X3_T := 0;
  End_If;
// ObjIdx=9 => Transition "t4"
  // Steps Below: id=0 => X0 ;
  If (t4) Then
    X0 := True;     X0_T := 0;
End_If;
```

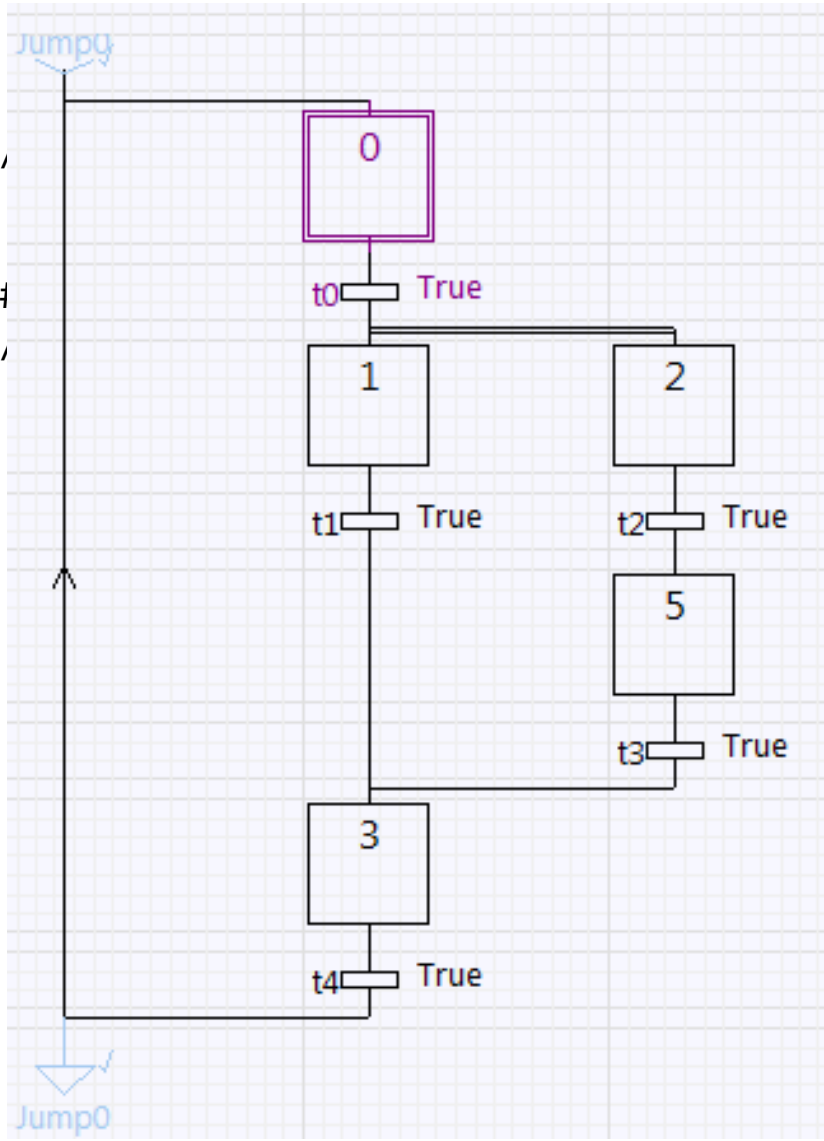


```
////////////////////////////////////////////////
///// If step active increment MW timer of step

  // 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;
  // ObjIdx=3 => Step "X2"
  If (s16) and (X2) Then
    X2_T := X2_T+1;
  end_if;
  // ObjIdx=6 => Step "X5"
  If (s16) and (X5) Then
    X5_T := X5_T+1;
  end_if;
  // ObjIdx=8 => Step "X3"
  If (s16) and (X3) Then
    X3_T := X3_T+1;
  end_if;
```

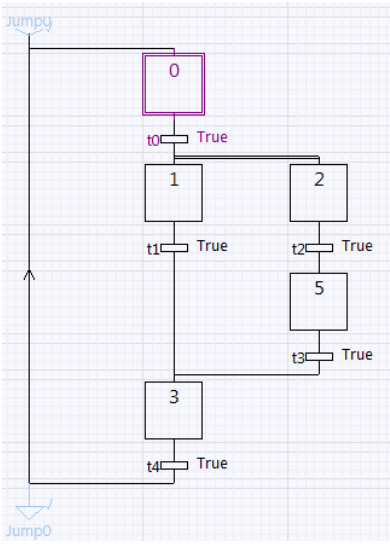


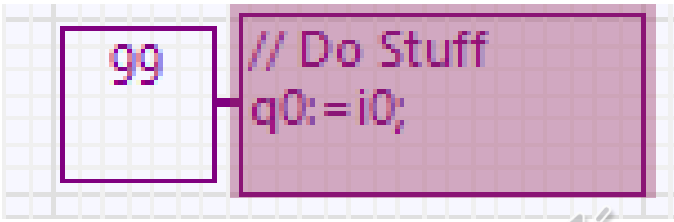


```
////////////////////////////////////////////////
//////// If step active,
//////// execute its action code
//######################## Page 0 ################
////////////////////////////////////////////////
```



```
////////////////////////////////////////////////////////////
//////// If step active, execute its action code ///////////
//######################## Page 0 ########################//
////////////////////////////////////////////////////////////

  // ObjIdx=21 => Step "X99" (code...)
  If X99 Then
    // Do Stuff
    q0:=i0;
  End_If;
```

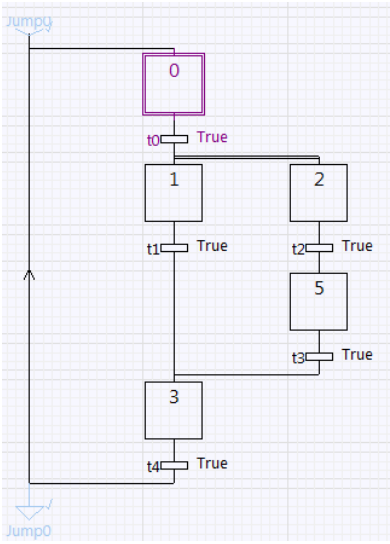


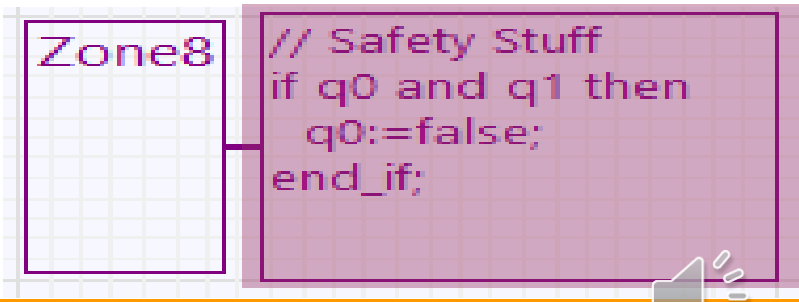


```
////////////////////////////////////////////////////////////
/////////////////////////// Zone8 ///////////////////////////
//######################### Page 0 ########################//
////////////////////////////////////////////////////////////
```

```
// Safety Stuff
if q0 and q1 then
   q0:=false;
end_if;
```

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



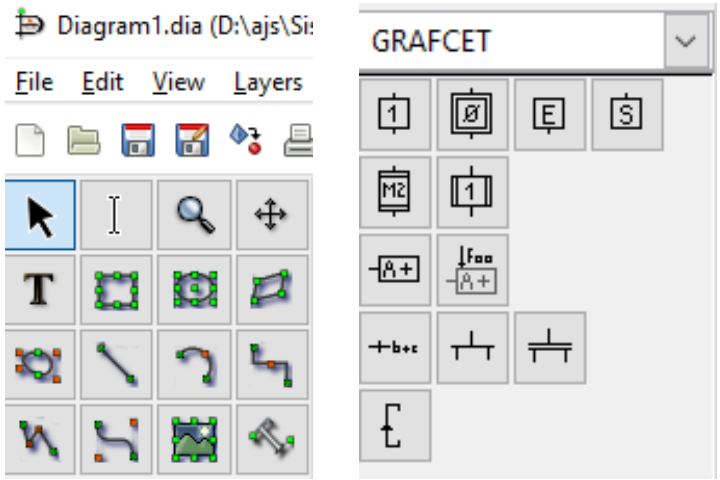


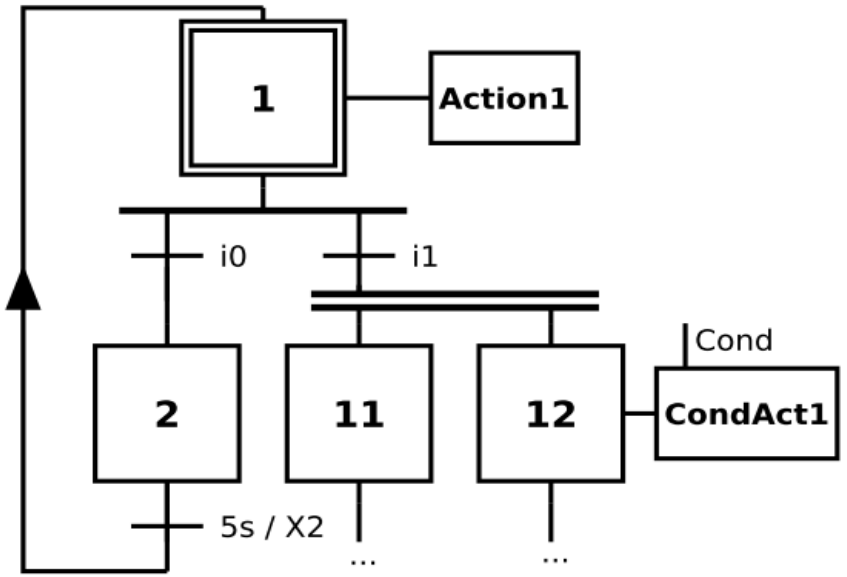
Comparação

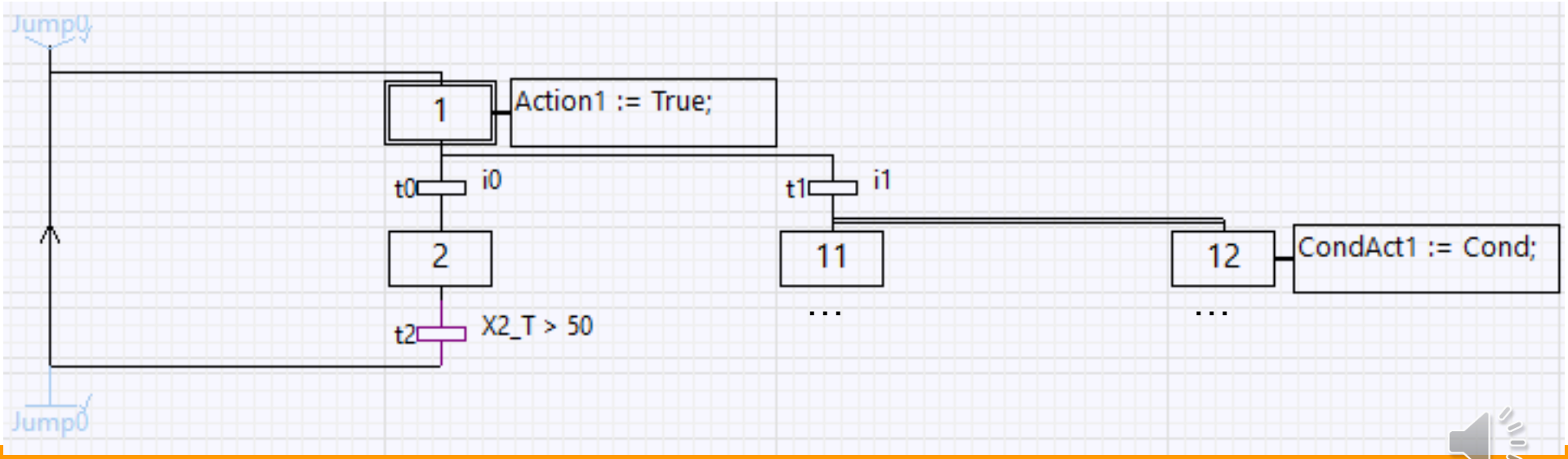
Norma

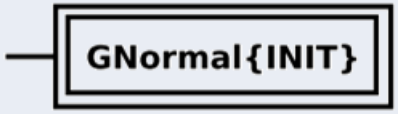
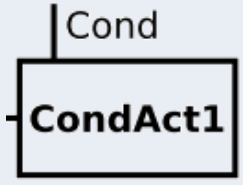
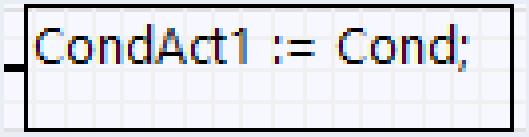
FEUPautom

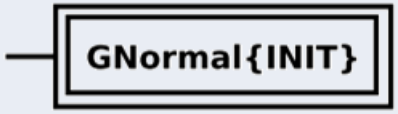
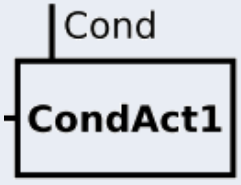
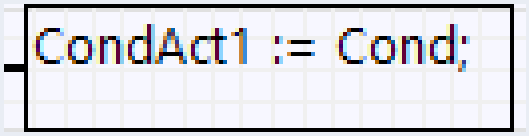
Norma 848 vs FEUPAutomGrafcet





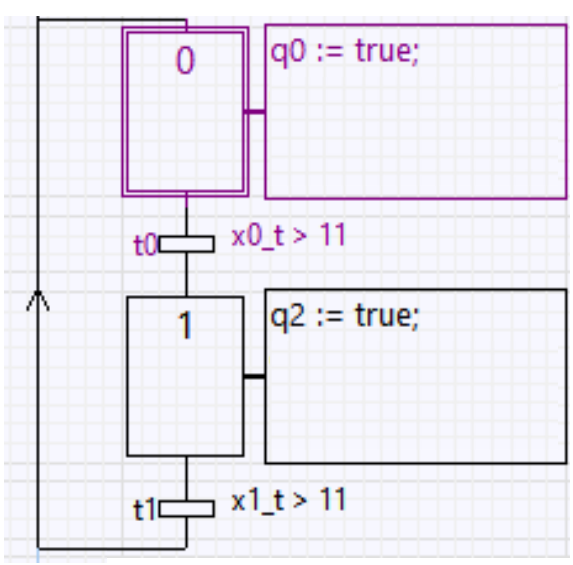


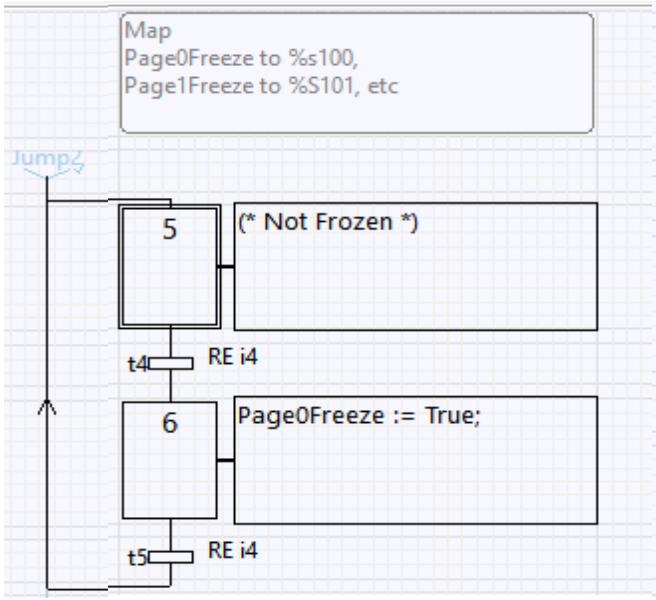
Norma Grafcet 848	Grafcet FEUPAutom *<5.50*
2 s / X5	X5_T > 20
	Manually reset all implicated steps (in a page with higher number / rank) $X_{all\_implicated}$:= false; ...
	Manually set and reset all steps involved (in a page with higher number / rank) X_{listed_steps} := true; ... X_{all_others} := false; ...
	Manually set and reset all steps involved (in a page with higher number / rank) $X_{initial\_steps}$:= true; ... X_{all_others} := false; ...
	Use code in the “Zone3” special step to manually reset transitions implicated in the freeze macro action $T_{all\_implicated}$:= false; ...
	

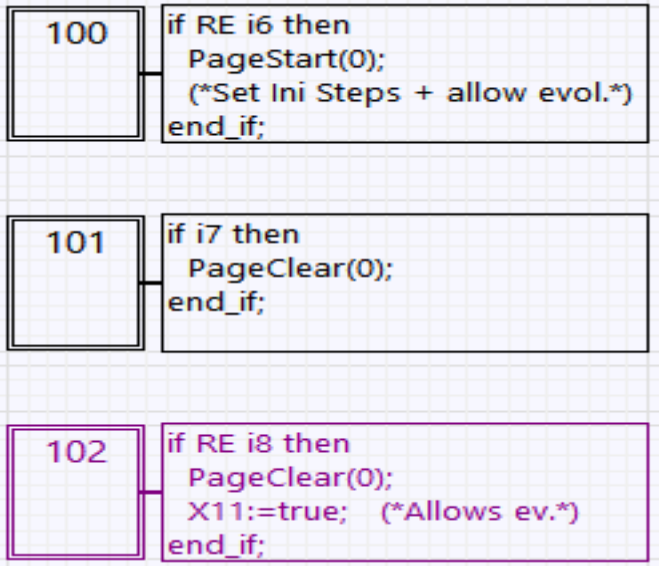
Norma Grafcet 848	Grafcet FEUPAutom *v5.50+*
2 s / X5	X5_T > 20
	In a page with higher number / rank **`PageClear(PageNumber)`**
	In a page with higher number / rank **`PageClear(PageNumber)`** and then set steps X_{listed_steps} **`:= true; ...`**
	In a page with higher number / rank **`PageStart(PageNumber)`**
	In a page with higher number / rank, use system bit Page0Freeze➔%s100, Page1Freeze➔%s101, etc
	

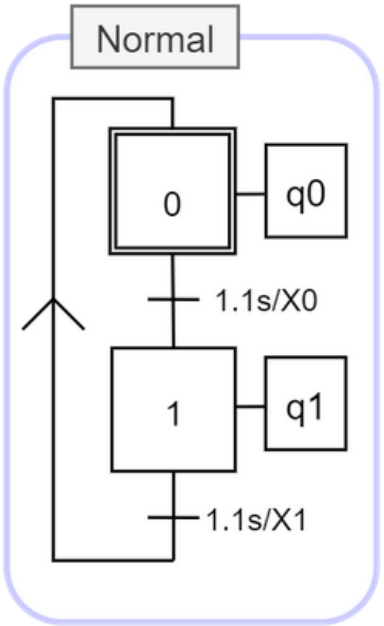
Freeze, PageStart/Page Clear

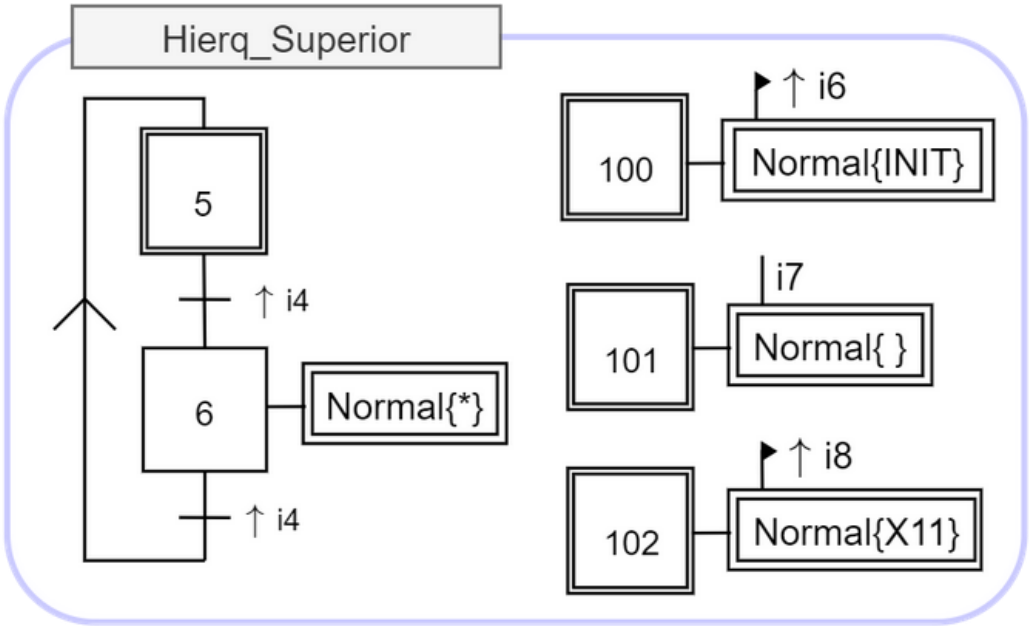
FEUPAutom v.5.50+











“FEUPAutom_Grafcet”
(incluindo Grafcet hierárquico)

Armando Sousa

- Fim -

27/04/2020