

Verificação Formal de Software - Exercícios 1

Sistemas de Transições e Comunicação entre Processos

1. Seja circuito sequencial com n bits de entrada x_1, \dots, x_n , y_1, \dots, y_m bits de saída e r_1, \dots, r_k registos de um bit. Os x_1, \dots, x_n têm valores não determinísticos em $\{0, 1\}$ cada um e os r_j têm valores iniciais $c_{0,j} \in \{0, 1\}$ para $1 \leq j \leq k$. Om sistema de transições que o representa é $TS = (S, Act, \longrightarrow, I, AP, L)$ tal que

- $S = Eval(x_1, \dots, x_n, r_1, \dots, r_k)$
- $I = \{(a_1, \dots, a_n, c_{0,1}, \dots, c_{0,k}) \mid a_1, \dots, a_n \in \{0, 1\}\}$
- $Act = \{\tau\}$
- $AP = \{x_1, \dots, x_n, y_1, \dots, y_m, r_1, \dots, r_k\}$
- $L : S \rightarrow 2^{AP}$. onde

$$L(a_1, \dots, a_n, c_1, \dots, c_n) = \{x_1 \mid a_i = 1\} \cup \{r_j \mid c_j = 1\} \cup \{y_i \mid s \models \lambda_{y_i}(a_1, \dots, a_n, c_1, \dots, c_k) = 1\},$$

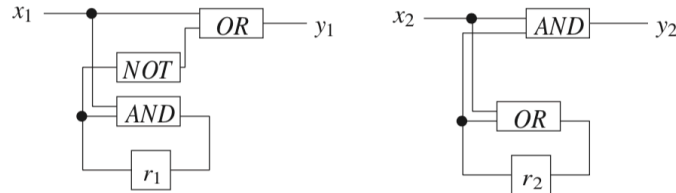
onde $\lambda_{y_i} : S \rightarrow \{0, 1\}$ é função booleana que é obtida no circuito para cada y_i .

As transições são determinadas pela funções de transição dos registos δ_{r_j} que resultam do circuito e tal que

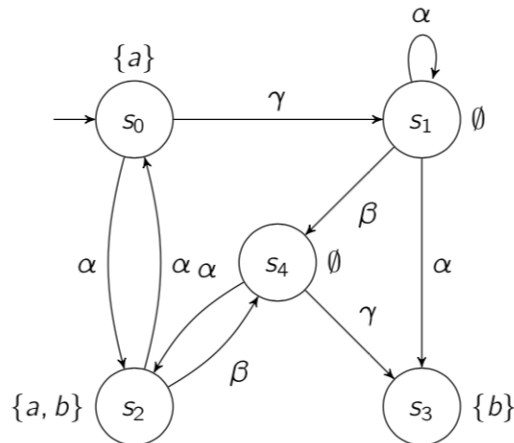
$$(a_1, \dots, a_n, c_1, \dots, c_k) \xrightarrow{\tau} (a'_1, \dots, a'_n, c'_1, \dots, c'_k),$$

onde $c'_j = \delta_{r_j}(a_1, \dots, a_n, c_1, \dots, c_k)$ e os a'_i são quais quer (decididos não-deterministicamente)

- (a) Determina os sistemas de transições associados aos seguintes circuitos sequenciais



2. Considera o seguinte sistema de transições T_1 :



- (a) Define formalmente T_1 .
- (b) Especifica uma execução finita e uma infinita em T_1 .
- (c) Indica se T_1 é *AP*-determinístico ou ações-determinístico.
3. Determinar o grafo de programa do seguinte fragmento de programa, indentificando as localizações no código e associando ações às respectivas instruções.
- (a) Supor no início $x = 2$
- ```

if $x > 2$ then
 $x := 0$
 $x := x + 1$

```
- (b) Supor no início  $x = 1$  e  $y = 2$ .
- ```

while  $x < 5$  do
   $x := x + 1$ 
   $y := y + x$ 

```
4. (Exclusão Mútua) O programa seguinte é um protocolo de exclusão mútua de dois processos de Pnuelli. Existe uma única variável s que pode tomar os valores 0 ou 1, tendo inicialmente o valor 1. Para além disso cada processo tem uma variável Booleana local y que inicialmente é 0. O programa para o processo P_i para $i = 1, 2$ é o seguinte:

```

while true do
  // seccao nao critica
   $(y_i, s) := (1, i)$ 
  wait until  $((y_{1-i} = 0) \vee (s \neq i))$ ;
  // seccao critica
   $y_i := 0$ 

```

onde $(y_i, s) := (1, i)$ é uma atribui 1 a y_i e i a s , num único passo (ação atómica).

Determina o grafo de programa de um processo P_i (considerando localizações diferentes para as secções não crítica e a crítica) e o correspondente sistema de transições.