MVS – Esercizi proposti - 6 marzo 2013

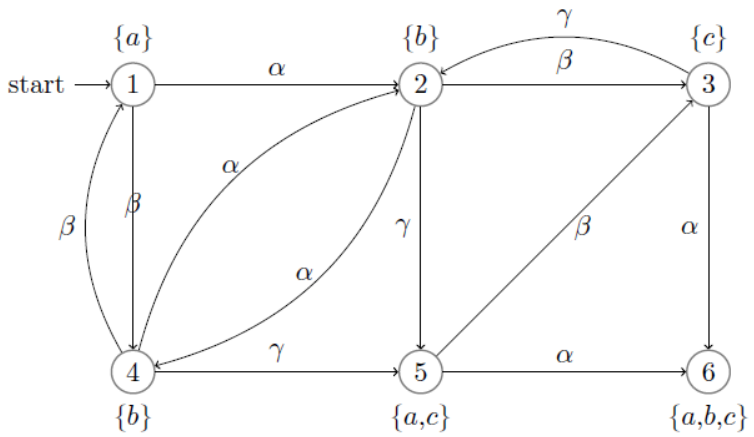
Esercizio 1

We consider the basic definitions for transition systems. Let TS be the transition system depicted below.

a) Give the formal definition of TS.

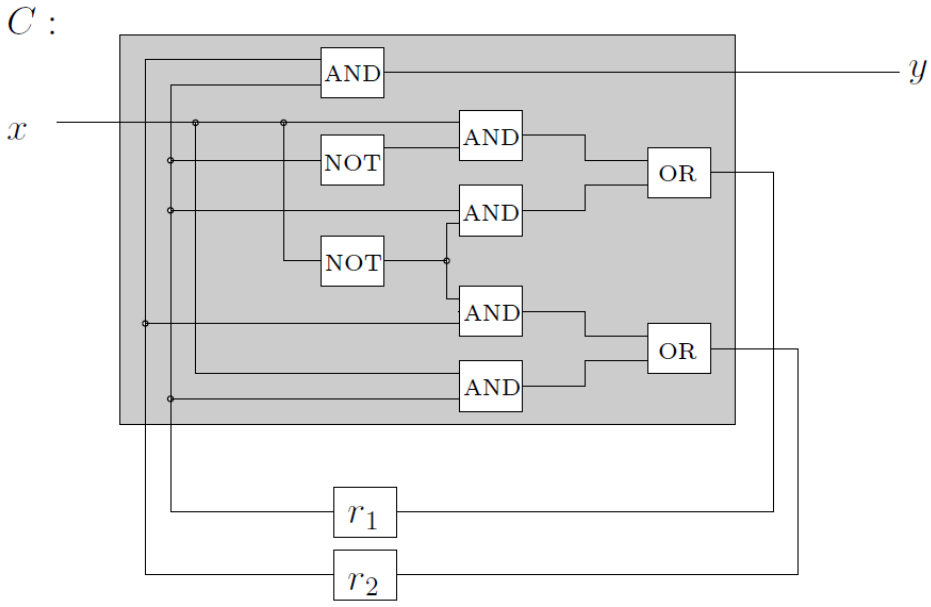
b) Specify a finite and an infinite execution of TS.

c) Decide whether TS is deterministic. Justify your answer!



Esercizio 2

Consider the following sequential hardware circuit:



Give the transition system representation TS of the circuit C.

Esercizio 3

Consider the following mutual exclusion algorithm that uses the shared variables y_1 and y_2 (which are initially both 0):

Process P_1:

while true **do**
  ... non-critical section ...
  $y_1 := y_2 + 1$
  wait until $(y_2 = 0) \vee (y_1 \leq y_2)$
  ... critical section ...
  $y_1 := 0$
  ... non-critical section ...
od

Process P_2:

while true **do**
  ... non-critical section ...
  $y_2 := y_1 + 1$
  wait until $(y_1 = 0) \vee (y_2 < y_1)$
  ... critical section ...
  $y_2 := 0$
  ... non-critical section ...
od

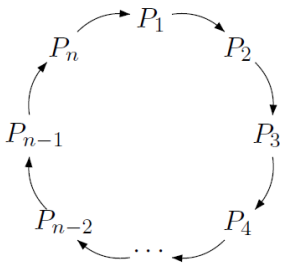
Questions:

a) Give the program graph representation of both processes.
(A pictorial representation suffices)

b) Give the reachable part of the transition system of $P_1 ||| P_2$ where $y_1 \leq 2$ and $y_2 \leq 2$.

Esercizio 4

Consider the following leader election algorithm: For $n \in \mathbb{N}$, n processes $P_1, \ldots, P_n$ are located in a ring topology where each process is connected by an unidirectional channel to its neighbour as outlined on the right.
To distinguish the processes, each process is assigned a unique identifier $id \in \{1, \ldots, n\}$. The aim is to elect the process with the highest identifier as the leader within the ring. Therefore each process executes the following algorithm:




```
send (id);                          initially set to process' id
while (true) do
    receive (m);
    if (m == id) then stop;         process is the leader
    if (m > id) then send (m);        forward identifier
od
```

a) Model the leader election protocol for n processes as a channel system.

b) Give an initial execution fragment of $TS([P_1|P_2|P_3])$ such that at least one process has executed the send-statement within the body of the while-loop.
Assume for $1 \leq i \leq 3$, that process P_i has identifier $id_i = i$.