

Specifica – parte IIB

Leggere Sez. 5.5.4, 5.5.4.1 Ghezzi et al.



Reti di Petri

- Adatte alla specifica di sistemi concorrenti:
 - Formalismo rigoroso e non ambiguo
 - Rappresentazione grafica relativamente semplice e intuitiva.



Reti di Petri

- Estensione degli automi a stati finiti, ma
- stato e transizione di stato sono concetti distribuiti:
 - Lo stato è dato da più stati parziali e indipendenti
 - Una transizione modifica solo una parte dello stato globale del sistema
 - Transizioni indipendenti possono avere luogo contemporaneamente



Reti di Petri

- Adatte alla rappresentazione di sistemi asincroni (in cui gli eventi non accadono a frequenze prestabilite)
- Nella versione più semplice, definiscono solo un ordinamento parziale fra eventi, astruendo dalla nozione di tempo.



Definizione: rete

- Una rete N è una quadrupla
$$N = (P, T, F, W)$$

dove

- P è detto *insieme dei posti*,
- T è detto *insieme delle transizioni*
- F è detta *relazione del flusso*
- W è detta *funzione peso*

Definizione: rete

- Condizioni:
 - $P \cup T \neq \emptyset$
 - $P \cap T = \emptyset$
 - $F \subseteq (P \times T) \cup (T \times P)$
 - $W: F \rightarrow \mathbb{N} \setminus \{0\}$ (valore di default 1)
- Marcatura (rappresenta lo stato):
 - $M: P \rightarrow \mathbb{N}$, assegna a ogni posto un numero naturale di *token* (o *gettoni*).

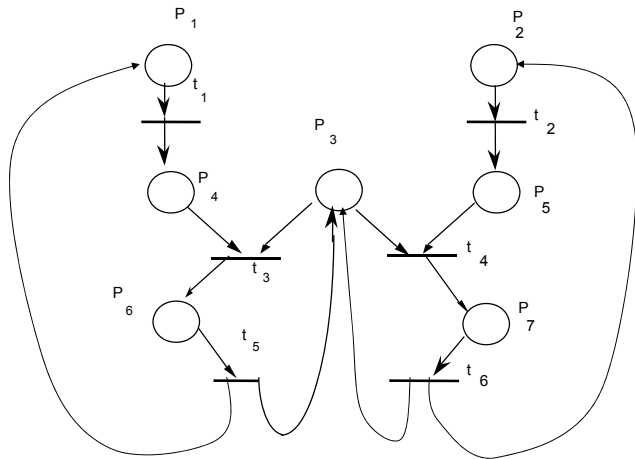
Marcatura iniziale

- Per stabilire l'evoluzione di una Rete di Petri è necessario fornire una *marcatura iniziale* M_0
- M_0 rappresenta l'insieme degli stati parziali, ossia lo stato globale, in cui la rete si trova all'inizio della sua evoluzione

Rappresentazione grafica

- Posti
- Transizioni
- Flusso: freccia da A e B se e solo se $(A, B) \in F$
- ↔ Può sostituire due frecce
- ⊙ Marcatura ($M(p) = 3$)

Esempio



Specifica 2B

9

Definizioni

- Un posto p è un *posto di input* di una transizione t se e solo se $(p,t) \in F$ (cioè se una freccia va da p a t).

$$Pre(t) = \{p \mid (p,t) \in F\}$$

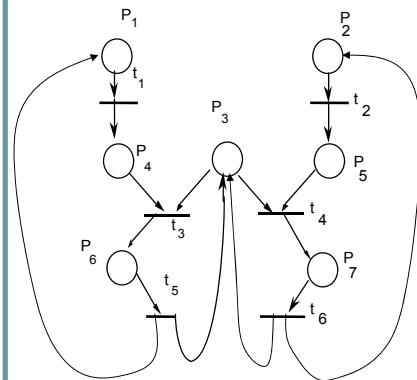
- Un posto p è un *posto di output* di una transizione t se e solo se $(t,p) \in F$ (cioè se una freccia va da t a p).

$$Post(t) = \{p \mid (t,p) \in F\}$$

Specifica 2B

10

Esempio



- $Pre(t_3) = \{p_3, p_4\}$
- $Post(t_3) = \{p_6\}$
- $Pre(t_5) = \{p_6\}$
- $Post(t_5) = \{p_1, p_3\}$

Specifica 2B

11

Transizione abilitata

- Una transizione t è *abilitata* se e solo se ognuno dei suoi posti di input è marcato con un numero di token maggiore o uguale al peso dell'elemento di flusso che lo unisce a t , cioè se e solo se

$$\forall p \in Pre(t) \quad M(p) \geq W((p,t))$$

Specifica 2B

12

Scatto di una transizione

- Una transizione abilitata può *scattare*.
- Lo scatto di una transizione t
 - rimuove da ognuno dei suoi posti p di input un numero di token pari al peso dell'elemento di flusso che unisce p a t
 - Aggiunge a ognuno dei suoi posti q di output un numero di token pari al peso dell'elemento di flusso che unisce t a q .
- Se più transizioni sono abilitate, può scattare una qualunque di esse (*non determinismo*).

Editor/simulatore di reti di Petri

- Platform Independent Petri Net Editor (PIPE)
- <http://pipe2.sourceforge.net>

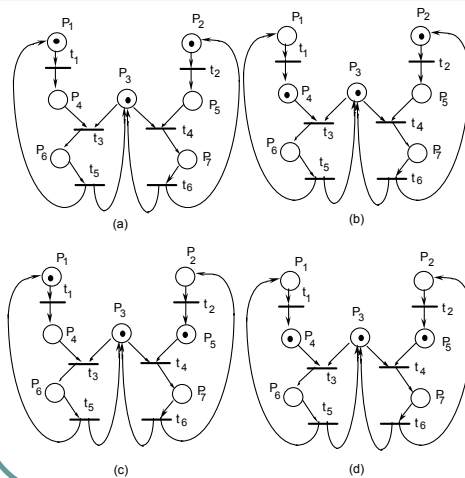
Esempio



Esempio

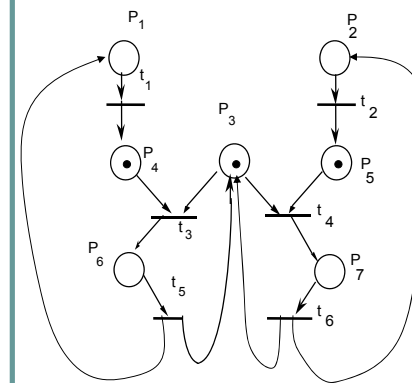


Esempio



- Dalla marcatura iniziale (a) si può passare a:
 - (b) se scatta t_1
 - (c) se scatta t_2 (una non esclude l'altra)
- Da (b) o da (c) si può passare a (d) o ad altri stati, es. se da (b) scatta t_3 .

Esempio

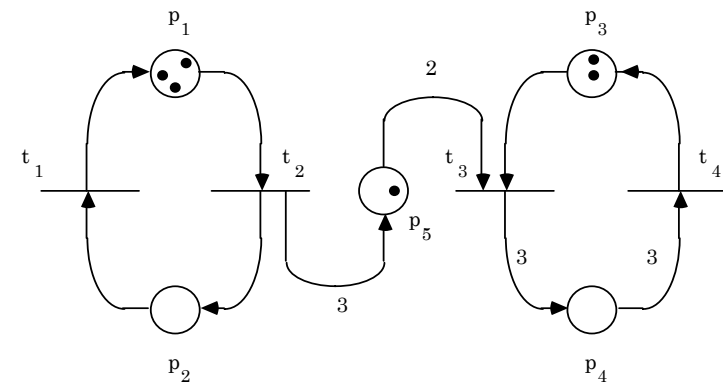


- In questo stato, t_3 e t_4 sono mutuamente esclusive (ognuna delle due, scattando, disabilita l'altra)
- Conflicto strutturale (un posto $p \in Pre(t_i)$ e $p \in Pre(t_j)$)
- Conflicto effettivo, marcatura insufficiente a far scattare sia t_i sia t_j

Sequenza di scatti

- Definizione: data una certa marcatura di una rete, una *sequenza di scatti* $t_1 t_2 \dots t_N$ è abilitata se:
 - $N=1$ e t_1 è abilitata, oppure
 - $N>1$, t_1 è abilitata e, data la marcatura risultante dallo scatto di t_1, t_2, \dots, t_{N-1} si ha che t_N è abilitata

Rappresentazione matriciale



Rappresentazione matriciale

$$O = \begin{array}{c|cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 3 & 0 \\ 0 & 3 & 0 & 0 \end{array} \quad I = \begin{array}{c|cccc} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 2 & 0 \end{array}$$

$$C = O - I = \begin{array}{c|cccc} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 3 & -3 \\ 0 & 3 & -2 & 0 \end{array}$$

$$m = \begin{array}{c|c} 3 \\ 0 \\ 2 \\ 0 \\ 1 \end{array} \quad \text{Scatta } t_2: \quad m' = \begin{array}{c|c} 2 \\ 1 \\ 2 \\ 0 \\ 4 \end{array}$$

Rappresentazione matriciale

• E' possibile stabilire l'abilitazione di una transizione in base alla sola matrice C nelle cosiddette **reti pure**

• Una rete pura è una rete P/T in cui
 $\forall t \in T, \text{Pre}(t) \cap \text{Post}(t) = \emptyset$

(si esclude che un posto sia d'ingresso e di uscita a una medesima transizione)

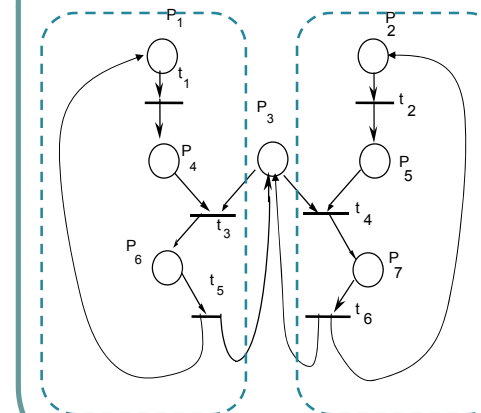
• Data una sequenza di scatti ammissibile, per queste reti:

$$m' = m + C s$$

Modellazione di sistemi concorrenti con reti di Petri

- Le marcature rappresentano lo stato in modo distribuito.
- Le transizioni rappresentano eventi.
- L'abilitazione di una transizione rappresenta il rispetto delle condizioni necessarie affinché l'evento corrispondente avvenga.

Esempio



- Due serie di attività indipendenti (t_1, t_3, t_5 e t_2, t_4, t_6)
- Accedono in modo mutuamente esclusivo a una risorsa comune (p_3).

Concorrenza

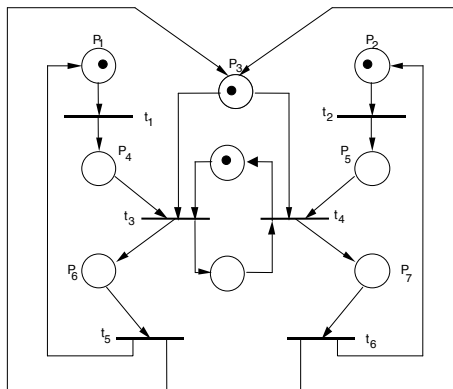
- Le transizioni abilitate possono scattare indipendentemente l'una dall'altra
- Condizioni comuni:
 - *Concorrenza*: due transizioni sono abilitate, e lo scatto di una non disabilita l'altra.
 - *Conflitto*: due transizioni sono abilitate, ma lo scatto di una disabilita l'altra (esempio: accesso a una risorsa condivisa).

Conflitto

- Le reti di Petri non prevedono meccanismi per risolvere automaticamente i conflitti
- Di due transizioni in conflitto, può scattare sempre la stessa
- L'attività che necessita dell'altra transizione è soggetta a *starvation*.

Per evitare la starvation

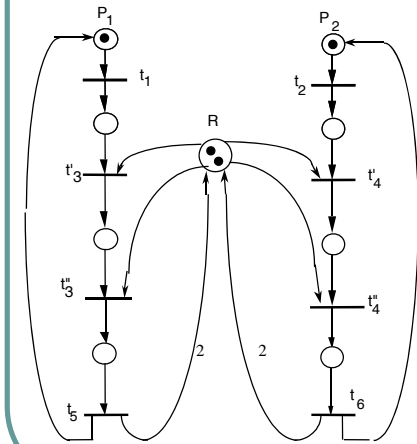
- Si può imporre l'alternanza:



Deadlock

- Un sistema è in deadlock se è in uno stato dal quale è impossibilitato a evolvere.
- Condizione da evitare, ma difficile da prevedere
- L'analisi delle reti di Petri permette di individuare i deadlock di un sistema.

Esempio

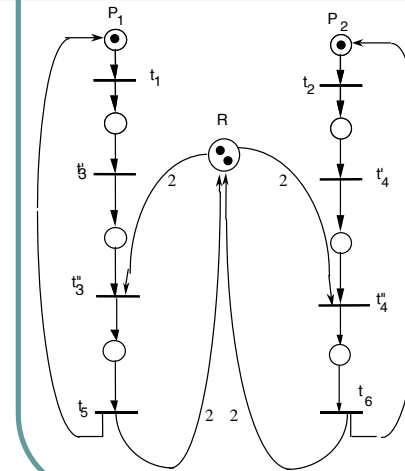


- Attività concorrenti che necessitano di due copie di una risorsa.
- Caso "normale": t_1, t_3', t_3'', t_5
- Caso che porta a deadlock: t_1, t_3', t_2, t_4'

Specifica 2B

29

Per evitare il deadlock

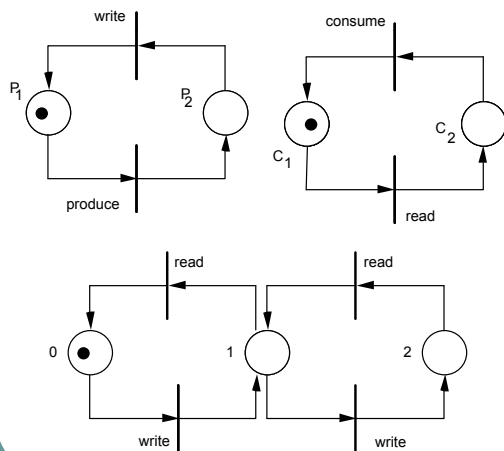


- Le due copie della risorsa vengono prelevate con una sola transizione

Specifica 2B

30

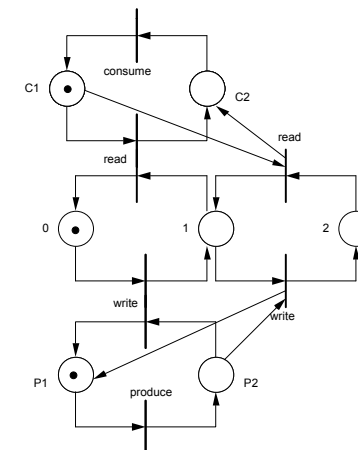
Esempio produttore / consumatore: reti separate



Specifica 2B

31

Esempio produttore / consumatore: rete unica



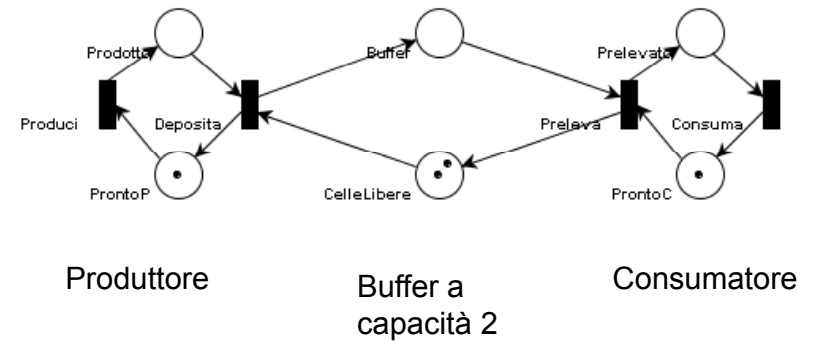
Specifica 2B

32

Posti a capacità limitata

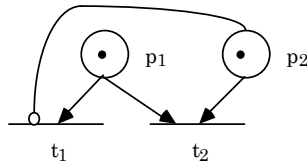
- E' possibile stabilire la capacità massima di un posto, cioè il numero massimo di token che può contenere.
- Non risultano abilitate le transizioni che porterebbero un posto ad superare la sua capacità massima.
- Si può simulare questo comportamento con un "posto complementare".

Esempio



Archi inibitori

- La presenza di un token in un posto di input disabilita la transizione



- t_1 è disabilitata, t_2 è abilitata

Proprietà di reti di Petri: raggiungibilità

- Una marcatura M' è raggiungibile da una marcatura M se esiste una sequenza di scatti che a partire da M produca M'
- Serve a verificare se si possono raggiungere stati indesiderati

Proprietà di reti di Petri: limitatezza

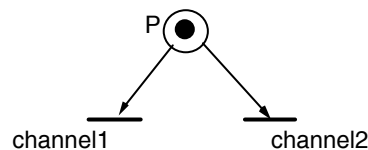
- Un posto si dice *k*-limitato se in qualunque marcatura raggiungibile non ha più di *k* token
- Una rete si dice limitata se tutti i suoi posti sono *k*-limitati per qualche *k*
- Una rete limitata è equivalente a un automa a stati finiti
- Qualsiasi rete può essere resa limitata con l'aggiunta di posti opportuni

Proprietà di reti di Petri: liveness

- Una transizione *t* è *live* se per ogni marcatura *M* raggiungibile, esiste una marcatura *M'* raggiungibile da *M* in cui *t* è abilitata
- Se almeno una transizione è *live*, non può esserci *deadlock* (la rete è *live*)

Limitazioni

- Es: vogliamo che un messaggio sia spedito attraverso uno di due canali:
 - channel₁ se il messaggio è ben formato
 - channel₂ altrimenti



- Ma i token non hanno valore

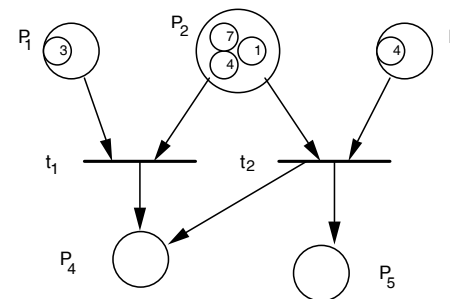
Estensione: assegnare valori ai token

- Alle transizioni sono associati predicati e funzioni
- I predicati, definiti sui valori dei token, determinano l'abilitazione delle transizioni
- Le funzioni determinano i valori dei token nei posti di output

Estensione: assegnare valori ai token

- Una transizione con N posti d'ingresso è abilitata se esistono N token, uno per ogni posto, i cui valori rispettano il predicato associato (*tupla pronta*).
- Possono esserci più tuple pronte
- La tupla pronta selezionata viene rimossa
- Si aggiungono token nei posti di output, con valori secondo le funzioni associate.

Esempio



$P2 > P1$
e
 $P4 := P2 + P1$
associati con $t1$

$P3 = P2$ e
 $P4 := P3 - P2$ e
 $P5 := P2 + P3$
associati con $t2$

Lo scatto di $t1$ con $\langle 3, 7 \rangle$ porta un token di valore 10 in $P4$. $t2$ allora può scattare con $\langle 4, 4 \rangle$

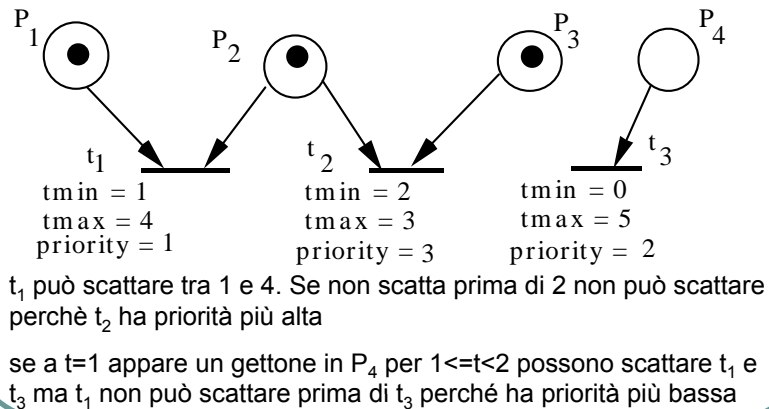
Reti con priorità

- Funzione $pri : T \rightarrow N$
- che assegna a ogni transizione un valore intero di priorità
- Fra più transizioni abilitate, scatta quella (o una di quelle) con priorità più alta
- Permette di esprimere politiche di scheduling (statiche; dinamiche se i token hanno un valore))

Reti temporizzate

- A ogni transizione associate due costanti $\langle t_{\min}, t_{\max} \rangle$
- t_0 nella marcatura iniziale
- Una transizione abilitata deve scattare fra t_{\min} e t_{\max} dopo l'istante di abilitazione.
- Se fra t_{\min} e t_{\max} viene disabilitata prima di scattare, non scatta.

Esempio: rete temporizzata con priorità

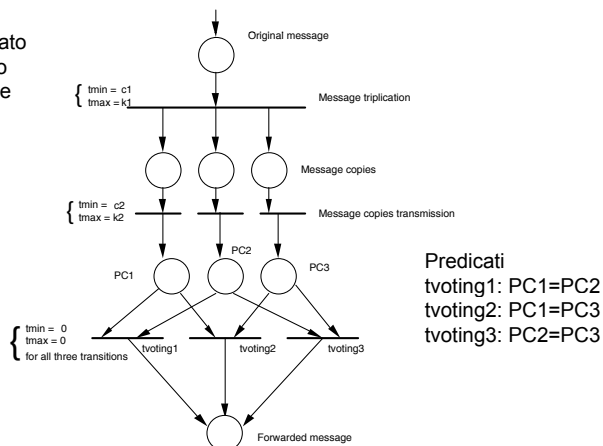


Esempio: replicazione e selezione di messaggi – 0

- Il messaggio dovrà essere triplicato. Le tre copie dovranno essere spedite attraverso tre canali fisici diversi. Il ricevitore dovrà accettare il messaggio sulla base di una politica di votazione "due su tre".

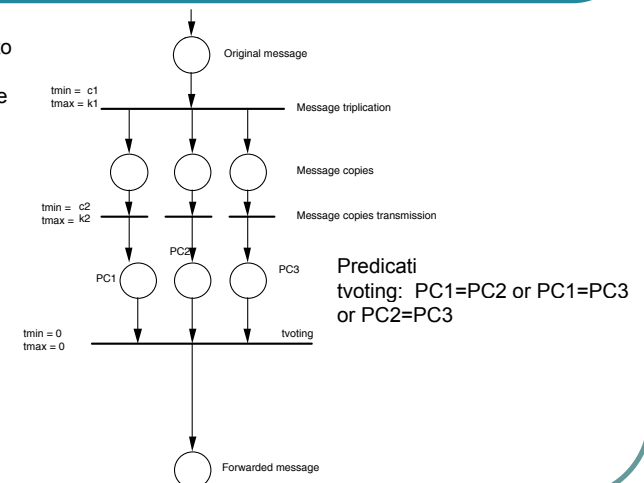
Esempio: replicazione e selezione di messaggi – 1

messaggio inoltrato non appena sono state ricevute due copie identiche



Esempio: replicazione e selezione di messaggi – 2

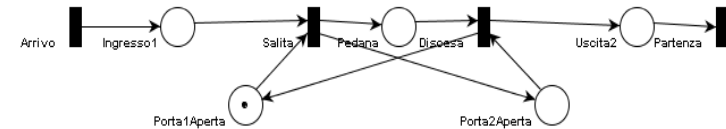
messaggio inoltrato solo quando sono state ricevute le tre copie



Esercizio

- Modellare con una Rete di Petri un sistema di controllo degli accessi costituito da una pedana e due porte. La pedana si trova tra le due porte. La prima porta è solitamente aperta e la seconda chiusa. Quando la pedana conta una persona, la prima porta si chiude e si apre la seconda porta.

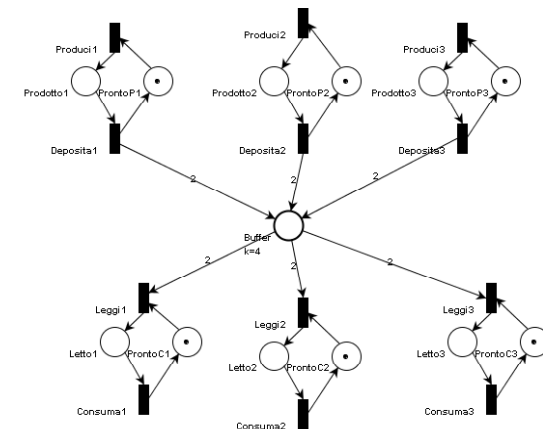
Soluzione



Esercizio

Modellare con una rete di Petri un sistema costituito da tre produttori, tre consumatori e un buffer di quattro celle (*realizzato come coda FIFO*). Ogni messaggio letto o scritto è pari a due celle del buffer. I produttori scrivono sul buffer (non pieno) o producono il messaggio, i consumatori leggono (dal buffer contenente almeno un messaggio) o elaborano il messaggio letto.

Soluzione



Soluzione

- Senza imporre un limite di capacità del posto che rappresenta il buffer:

