EXERCISE 4.2. Let $n \geqslant 1$. Consider the language $\mathcal{L}_n \subseteq \Sigma^*$ over the alphabet $\Sigma = \{A, B\}$ that consists of all finite words where the symbol B is on position n from the right, i.e., $\mathcal{L}$ contains exactly the words $A_1 A_2 \ldots A_k \in \{A, B\}^*$ where $k \geqslant n$ and $A_{k-n+1} = B$. For instance, the word $ABBAABAB$ is in $\mathcal{L}_3$.

(a) Construct an NFA $\mathcal{A}_n$ with at most $n{+}1$ states such that $\mathcal{L}(\mathcal{A}_n) = \mathcal{L}_n$.

(b) Determinize this NFA $\mathcal{A}_n$ using the powerset construction algorithm.

EXERCISE 4.3. Consider the transition system TS_{Sem} for the two-process mutual exclusion with a semaphore (see Example 2.24 on page 43) and TS_{Pet} for Peterson's algorithm (see Example 2.25 on page 45).

(a) Let P_{safe} be the regular safety property "process 1 never enters its critical section from its noncritical section (i.e., process 1 must be in its waiting location before entering the critical section)" and $AP = \{ wait_1, crit_1 \}$.

 (i) Depict an NFA for the minimal bad prefixes for P_{safe}.

 (ii) Apply the algorithm in Section 4.2 to verify $TS_{Sem} \models P_{safe}$.

(b) Let P_{safe} be the safety property "process 1 never enters its critical section from a state where $x = 2$" and $AP = \{ crit_1, x = 2 \}$.

 (i) Depict an NFA for the minimal bad prefixes for P_{safe}.

 (ii) Apply the algorithm in Section 4.2 to verify $TS_{Pet} \not\models P_{safe}$. Which counterexample is returned by the algorithm?

EXERCISE 4.1. Let $AP = \{a, b, c\}$. Consider the following LT properties:

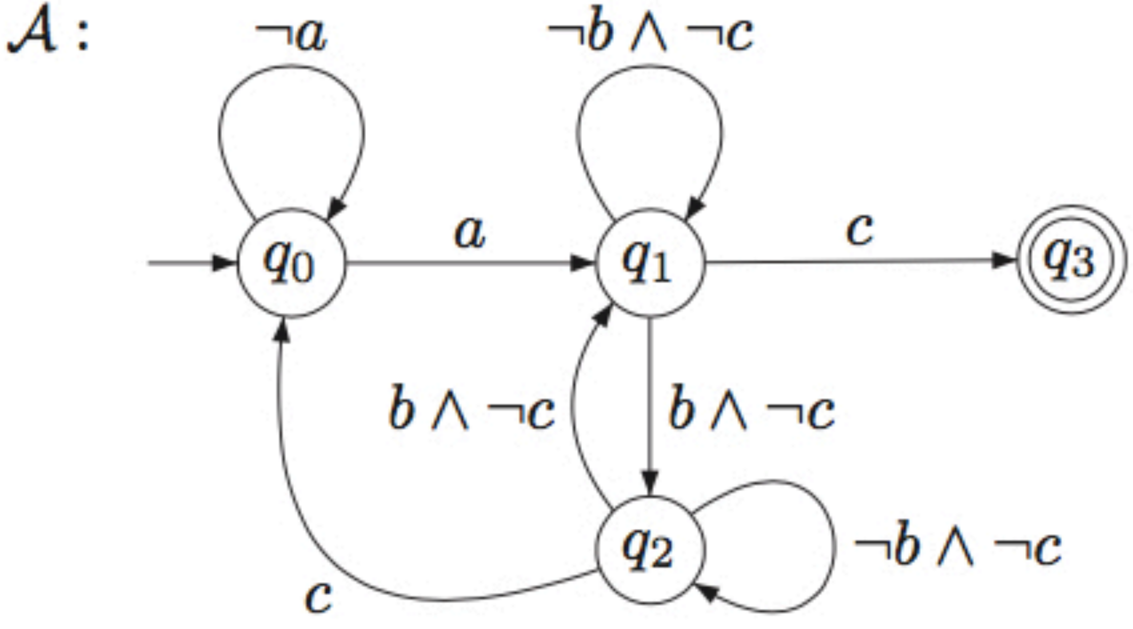
(a) If a becomes valid, afterward b stays valid ad infinitum or until c holds.

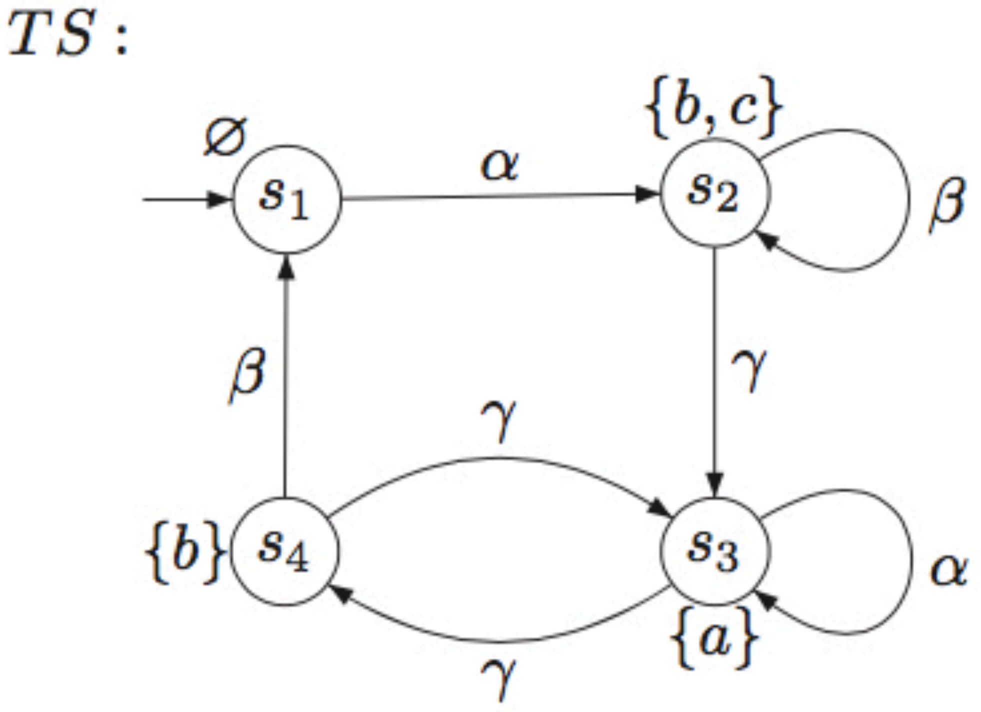
(b) Between two neighboring occurrences of a, b always holds.

(c) Between two neighboring occurrences of a, b occurs more often than c.

(d) $a \wedge \neg b$ and $b \wedge \neg a$ are valid in alternation or until c becomes valid.

For each property P_i $(1 \leqslant i \leqslant 4)$, decide if it is a regular safety property (justify your answers) and if so, define the NFA $\mathcal{A}_i$ with $\mathcal{L}(\mathcal{A}_i) = BadPref(P_i)$. (*Hint: You may use propositional formulae over the set AP as transition labels.*)





Construct the product $TS \otimes \mathcal{A}$ of the transition system and the NFA.

EXERCISE 4.15. Let $AP = \{a, b, c\}$. Depict an NBA for the LT property consisting of the infinite words $A_0 A_1 A_2 \ldots (2^{AP})^\omega$ such that

$$\forall j \geqslant 0.\, A_{2j} \models (a \ \vee \ (b \wedge c))$$

Recall that $A \models (a \ \vee \ (b \wedge c))$ means $a \in A$ or $\{b, c\} \subseteq A$, i.e., $A \in \{\{a\}, \{b, c\}, \{a, b, c\}\}$.

EXERCISE 4.18. Provide an example for a regular safety property P_{safe} over AP and an NFA $\mathcal{A}$ for its minimal bad prefixes such that

$$\mathcal{L}_\omega(\mathcal{A}) \neq (2^{AP})^\omega \setminus P_{safe}$$

when $\mathcal{A}$ is viewed as an NBA.

EXERCISE 4.27. Consider the transition systems TS_{Sem} and TS_{Pet} for mutual exclusion with a semaphore and the Peterson algorithm, respectively. Let P_{live} be the following ω-regular property over $AP = \{ wait_1, crit_1 \}$:

"whenever process 1 is in its waiting location then it will eventually enter its critical section"

(a) Depict an NBA for P_{live} and an NBA $\bar{\mathcal{A}}$ for the complement property $\bar{P}_{live} = (2^{AP})^{\omega} \setminus P_{live}$.

(b) Show that $TS_{Sem} \not\models P_{live}$ by applying the techniques explained in Section 4.4:

(i) Depict the reachable fragment of the product $TS_{Sem} \otimes \bar{\mathcal{A}}$

(ii) Sketch the main steps of the nested depth-first search applied to $TS_{Sem} \otimes \bar{\mathcal{A}}$ for the persistence property "eventually forever $\neg F$" where F is the acceptance set of $\bar{\mathcal{A}}$. Which counterexample is returned by Algorithm ?8

(c) Apply now the same techniques (product construction, nested DFS) to show that $TS_{Pet} \models P_{live}$.

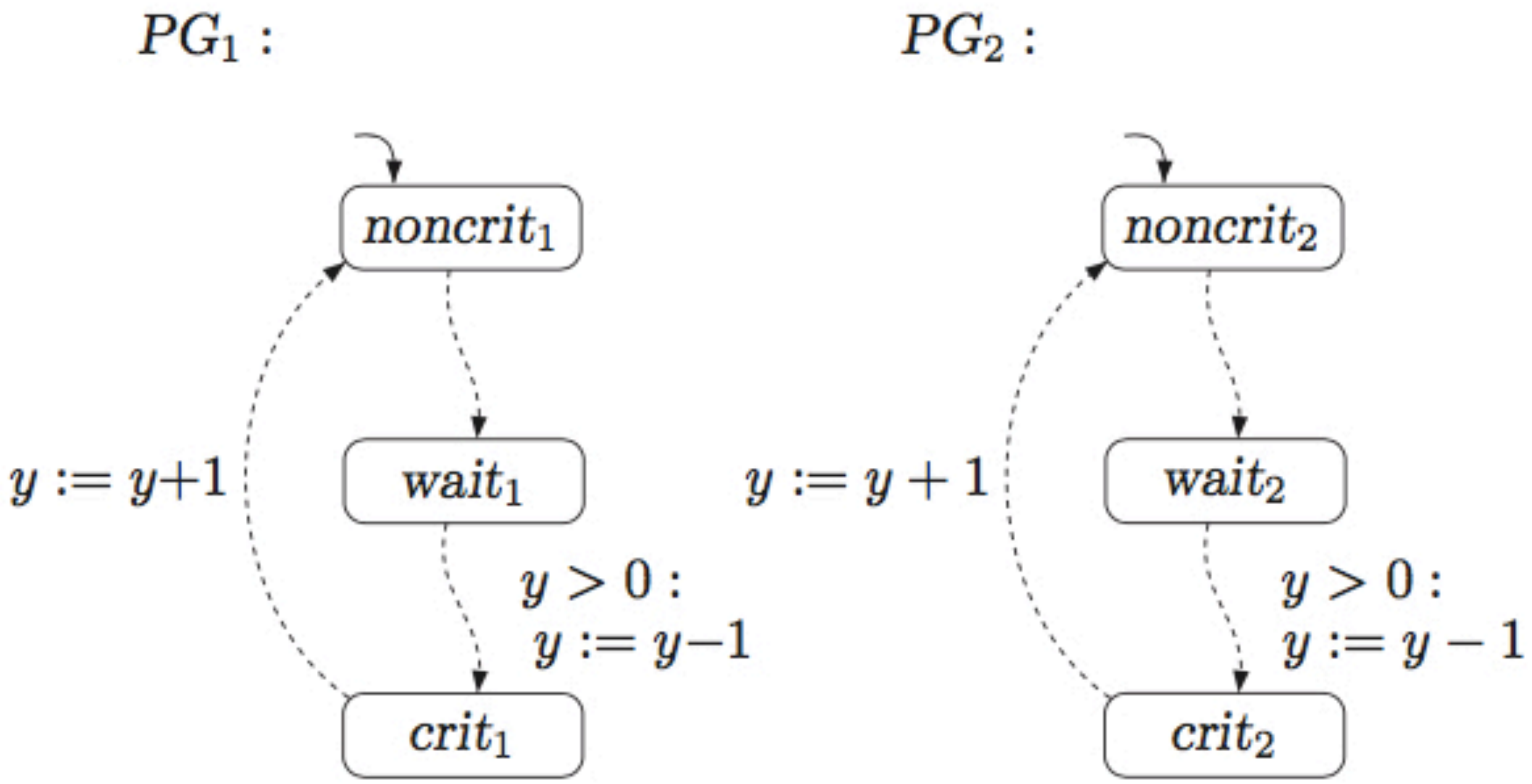


Figure 2.6: Individual program graphs for semaphore-based mutual exclusion.

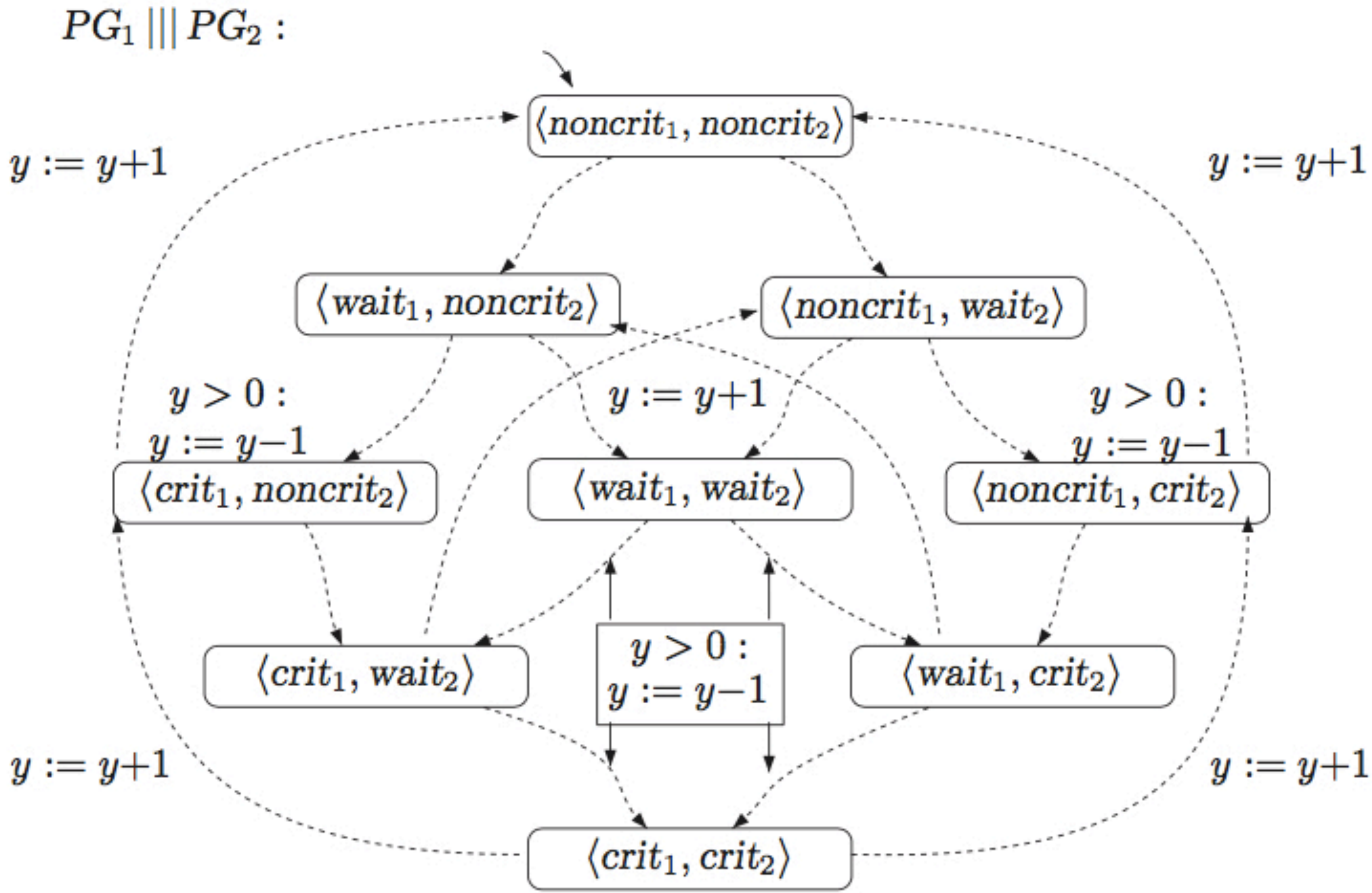


Figure 2.7: $PG_1 \,|||\, PG_2$ for semaphore-based mutual exclusion.

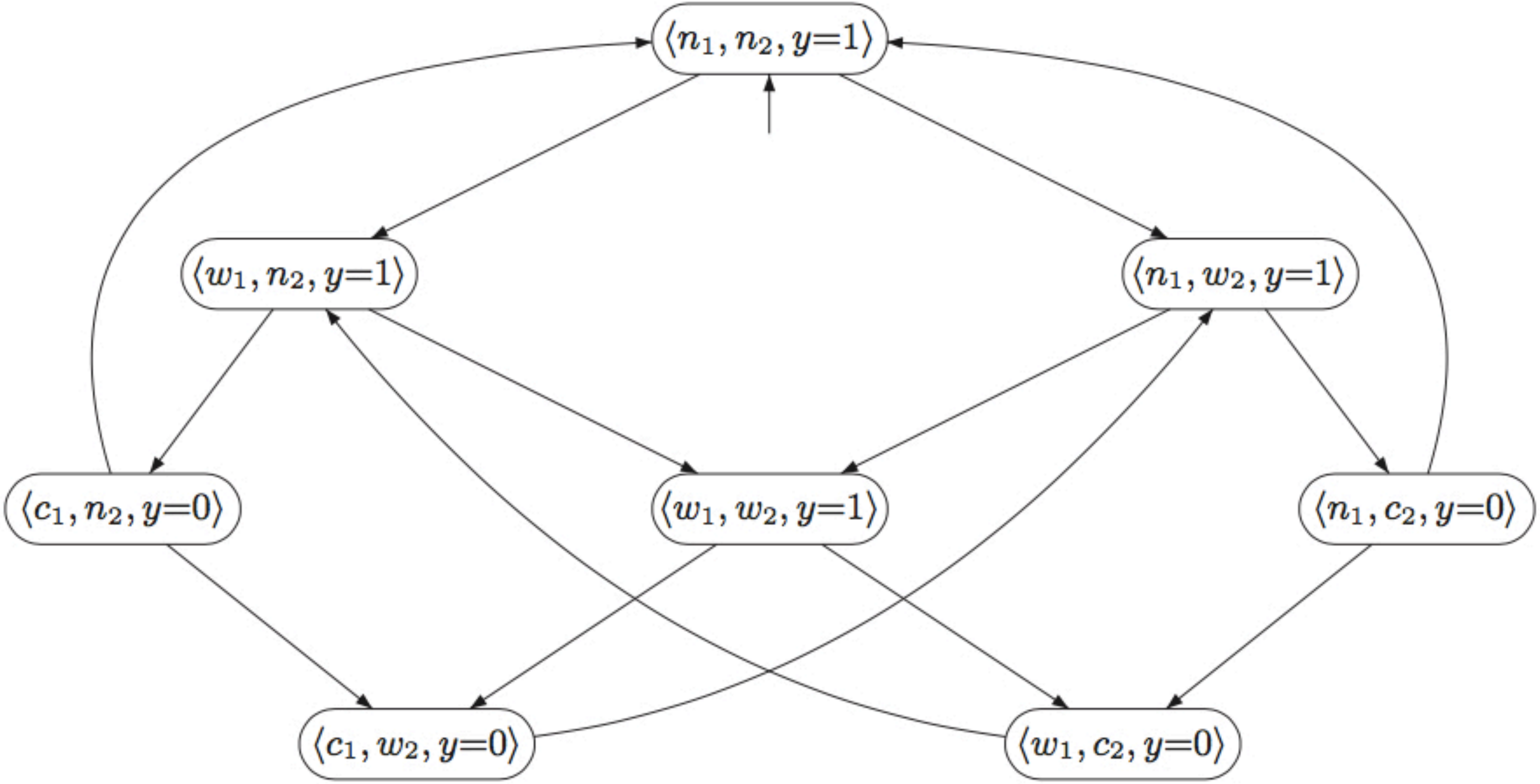


Figure 2.8: Mutual exclusion with semaphore (transition system representation).

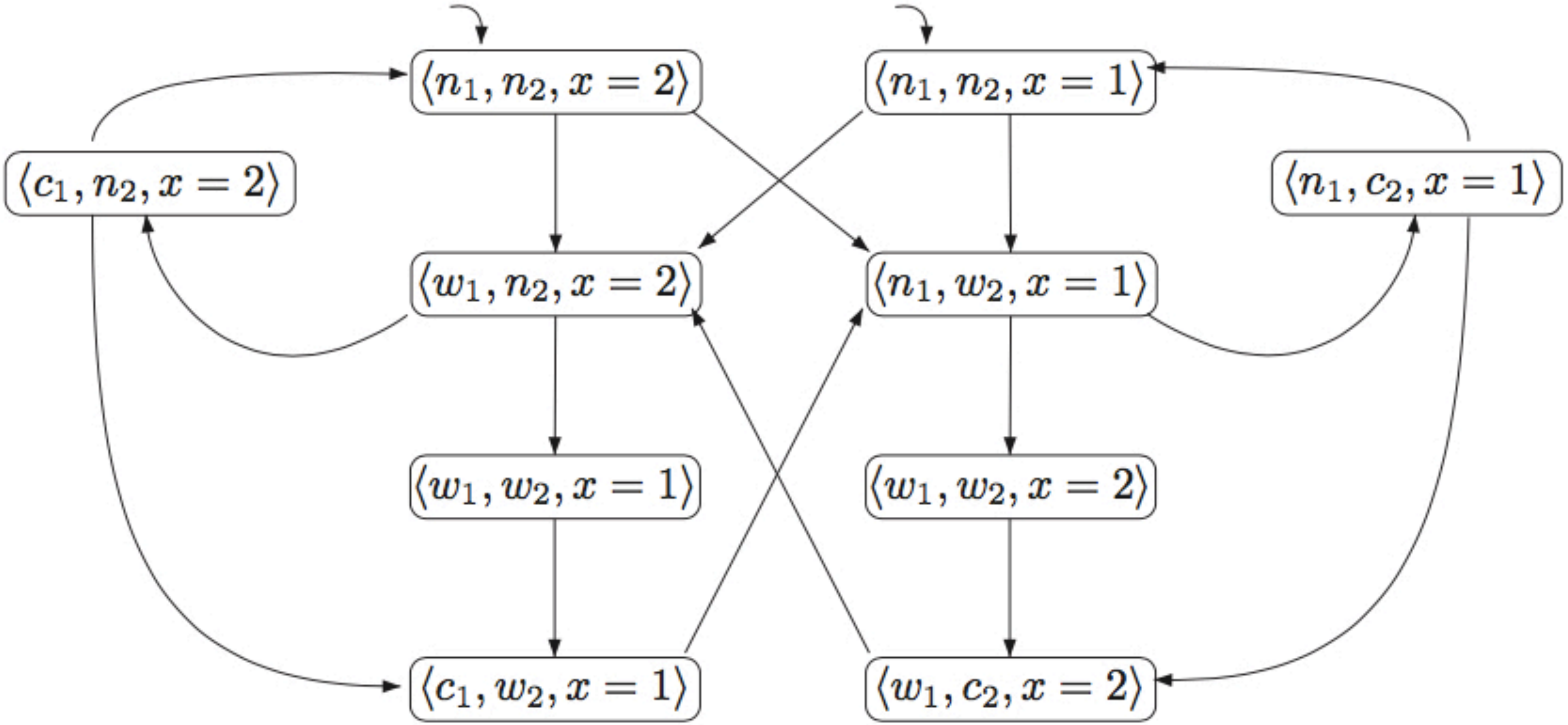


Figure 2.10: Transition system of Peterson's mutual exclusion algorithm.

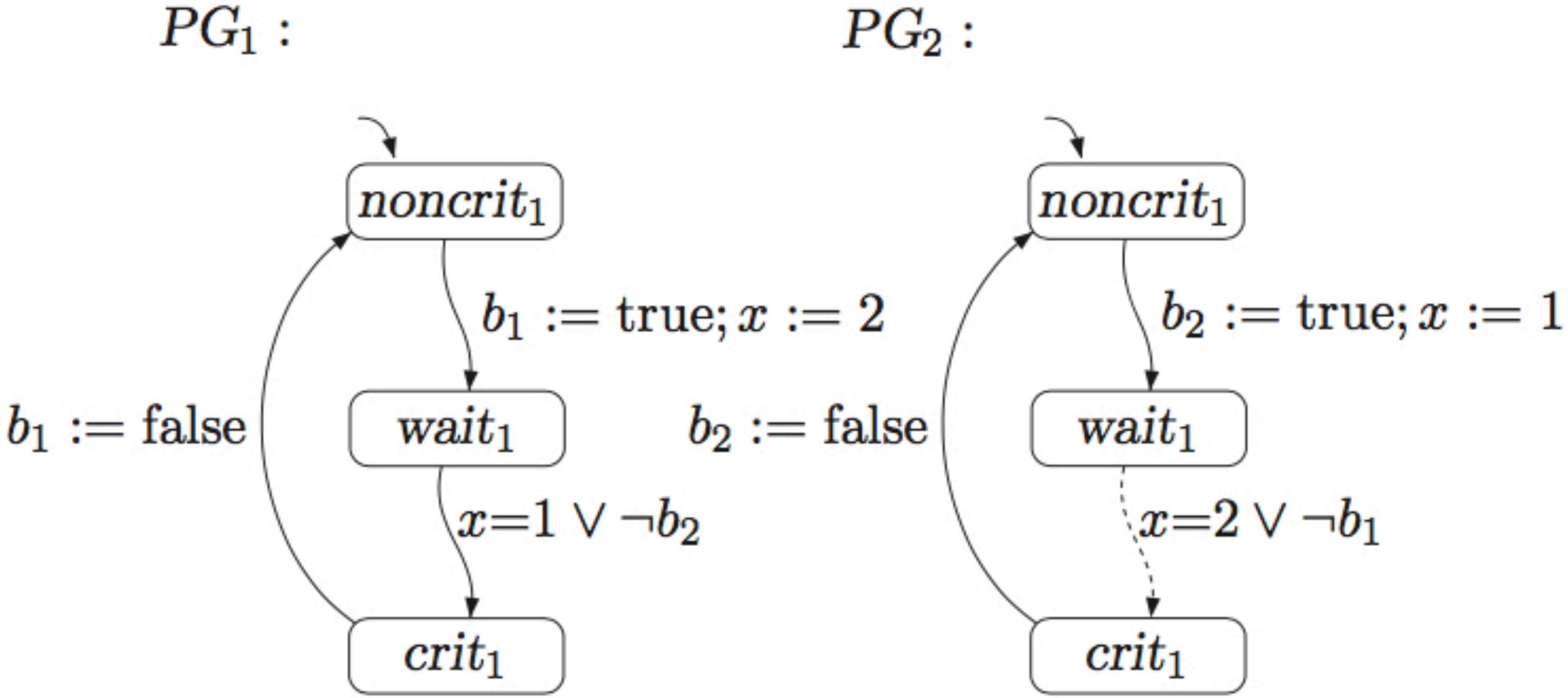


Figure 2.9: Program graphs for Peterson's mutual exclusion algorithm.

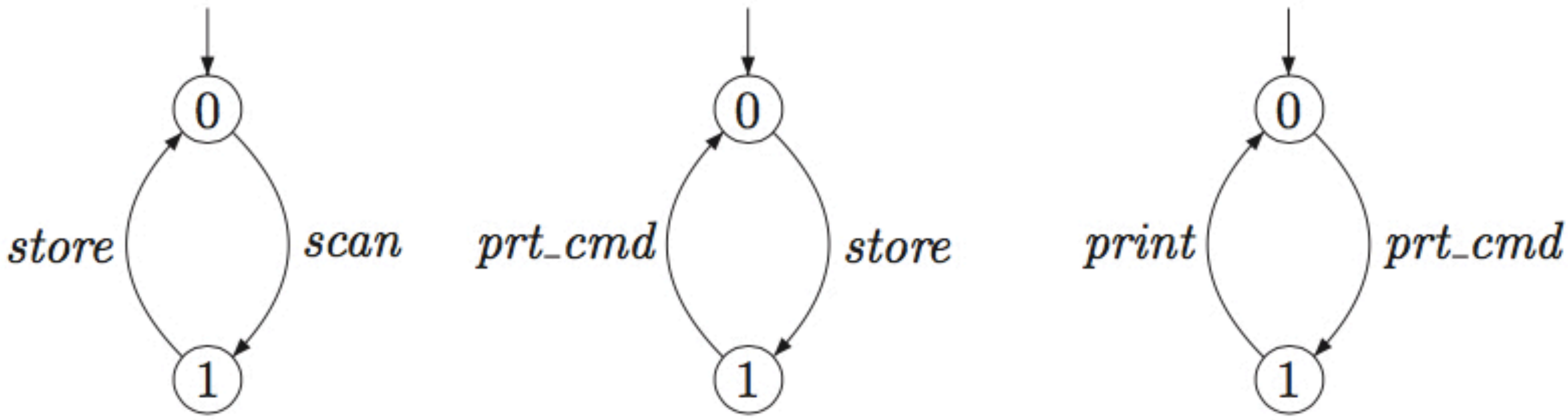


Figure 2.13: The components of the book keeping example.

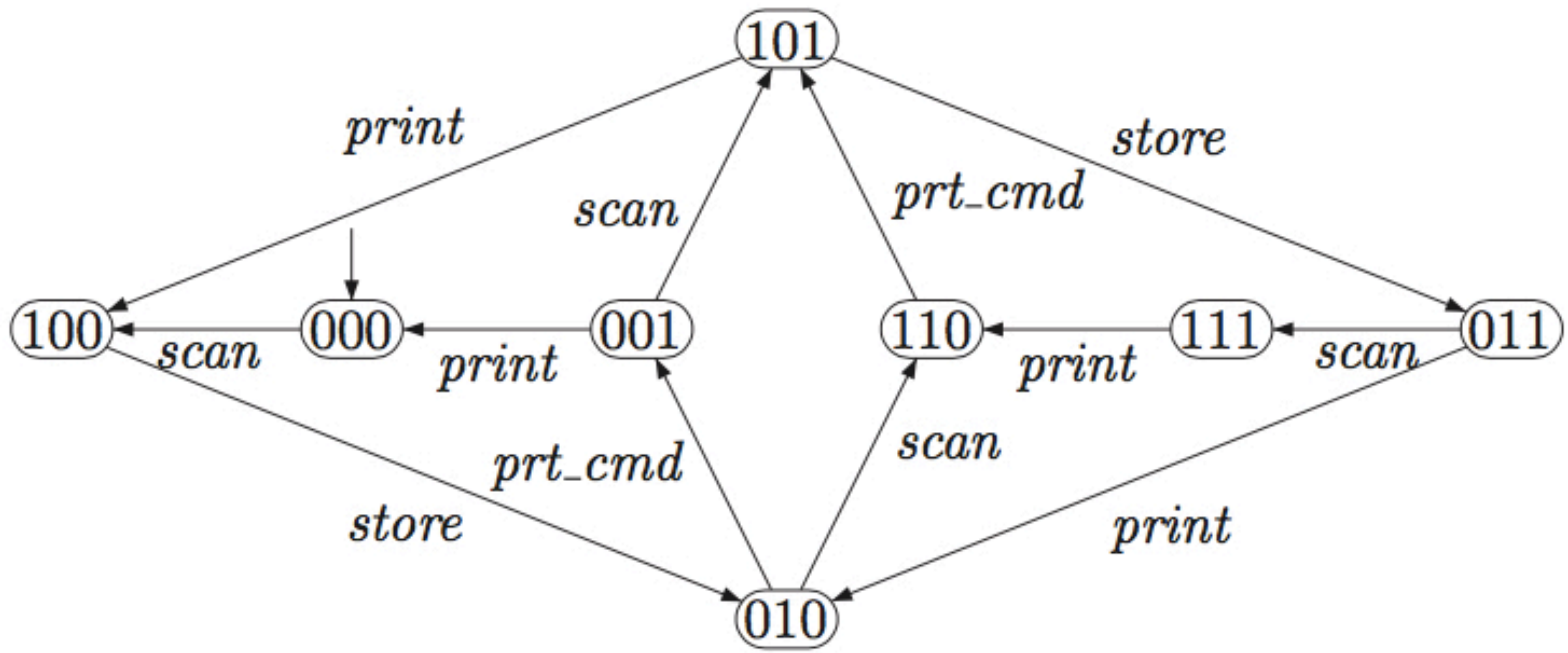


Figure 2.14: Transition system representation of the booking system.

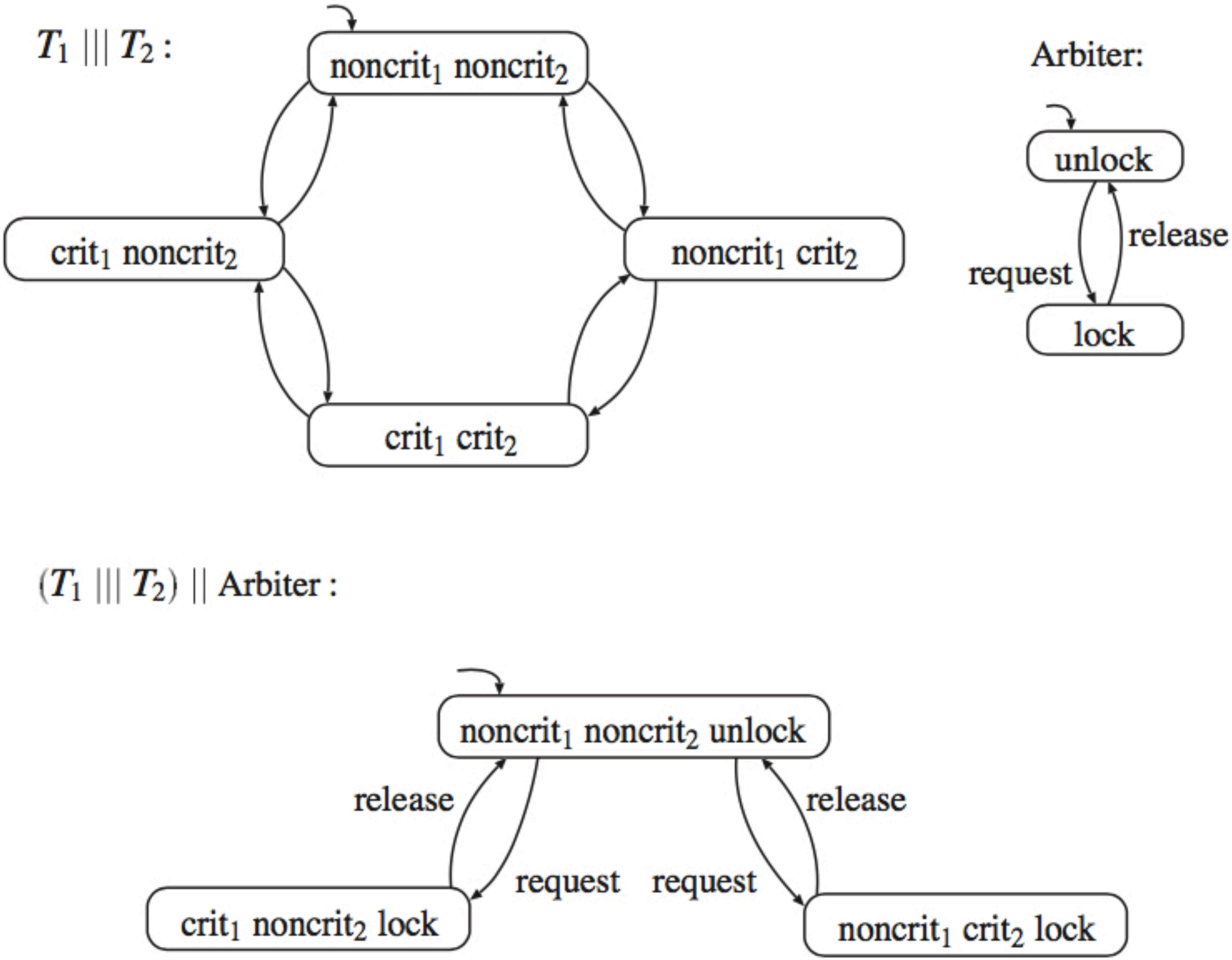


Figure 2.12: Mutual exclusion using handshaking with arbiter process.