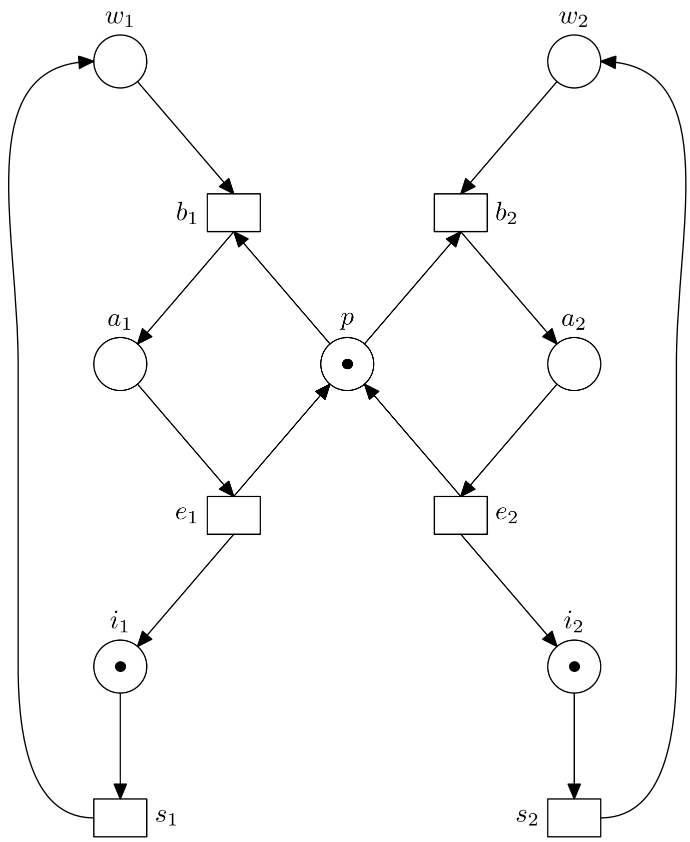
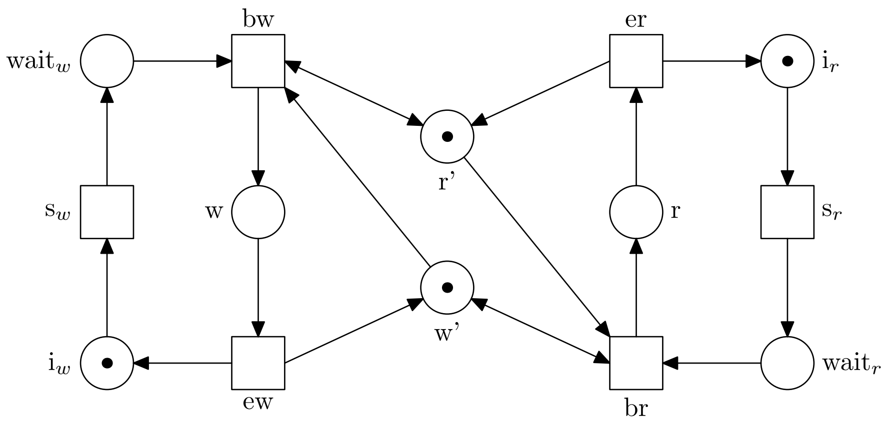
1. Below is presented a place/transition system. In the system, there are two processes, 1 and 2, using a shared resource. The place p controls the use of the resource. The system could, for example, model a situation where two processes need access to a shared file. Places a_i model the situation where one of the processes is accessing the file. Places i_i model the situation where the processes are doing something that does not need access to the file. Places w_i model the situation where a process is waiting for the access to the file.



   Show with invariants that there is mutual exclusion between processes, that is, the places a_1 and a_2 never have a token at the same time. Calculate also all invariants of the system.

2. In the next page a *state testing mutex algorithm* is presented. In the algorithm there are two processes, a reader and a writer. Only one process at a time can read or write (places r and w). Places r' and w' can be thought of as flags enabling the process to read or write. Using structural analysis, show that there is mutual exclusion between the processes.



Because the property can not be checked by using only invariants, we define a notion of *trap*:

Definition: trap. *Let* $N = \langle S, T, F, W, M_0 \rangle$ *be a place/transition system and let* $P \subseteq S$.

(a) P *is a trap iff* $P \neq \emptyset$ *and* $P^\bullet \subseteq {}^\bullet P$.

(b) P *is initialized iff* $\sum_{p \in P} M_0(p) > 0$.

Definition: trap inequality. *Let* $N = \langle S, T, F, W, M_0 \rangle$ *be a place/transition system,* $M(p)$ *the marking of place p and let* $P = \{p_1, \ldots, p_k\}$ *be a trap in* N. *Then*

$$M(p_1) + \cdots + M(p_k) \geq 1$$

is an inequality of P. *The inequality of* P *holds if the trap is initialized.*