10. What is the way of hierarchical nets construction in CPN Tools?

Laboratory lesson 6

Construction and investigation of models for automatic control

Goal of lesson: master the skill of construction and debugging of automatic control systems represented by Petri nets.

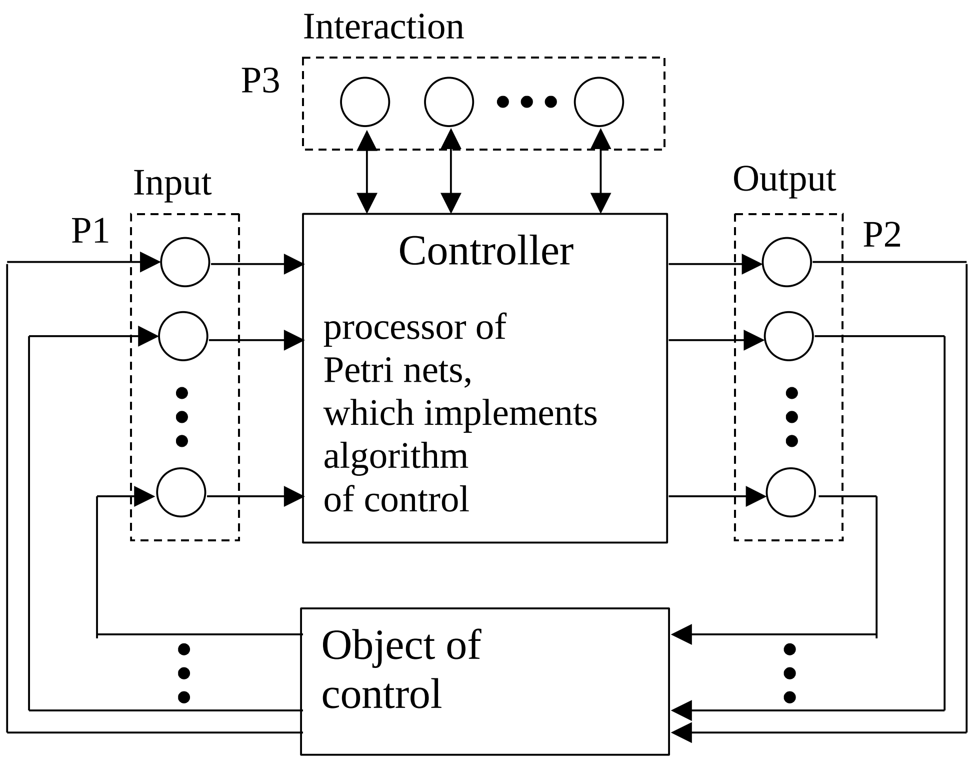
Preparation: for completion of the lesson, it is required to know basic classes of Petri nets, master the attainments of models' construction, study the description of automatic control system.

Order of lesson's execution

1. Study the model of automatic control of robot-manipulator.
2. Construct the corresponding Petri net into the environment of simulation system.
3. Execute an imitation of net's dynamics.
4. Determine the basic properties of net.
5. Construct the models of given control system and object of control.
6. Execute standalone debugging of control system.
7. Execute the complex debugging of control system together with the model of controlling object.
8. Investigate the properties of complex model.

Guidelines to lesson's completion

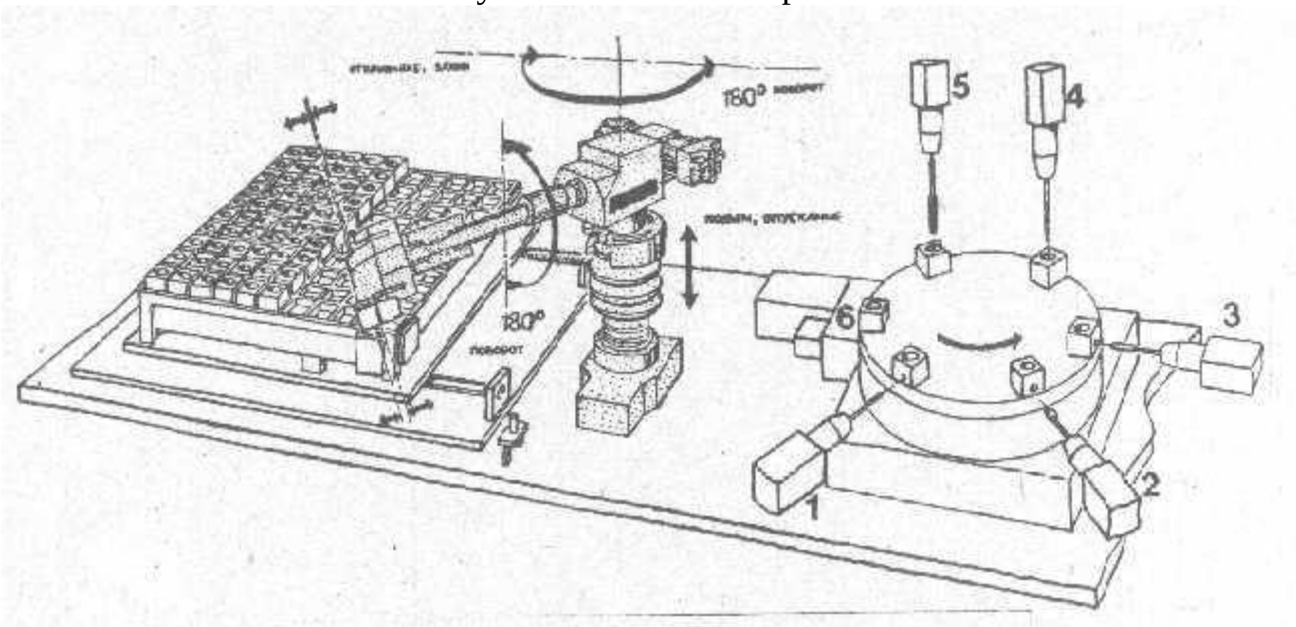
Let us consider the peculiarities of construction of automatic control systems based on a controller, which contains a processor of Petri nets (hardware of software):



Input places (P1) are directly connected with sensors of controlled object, output places (P2) are connected to executing devices (actuators); places P3 provides connection with operator and interaction with external control systems.

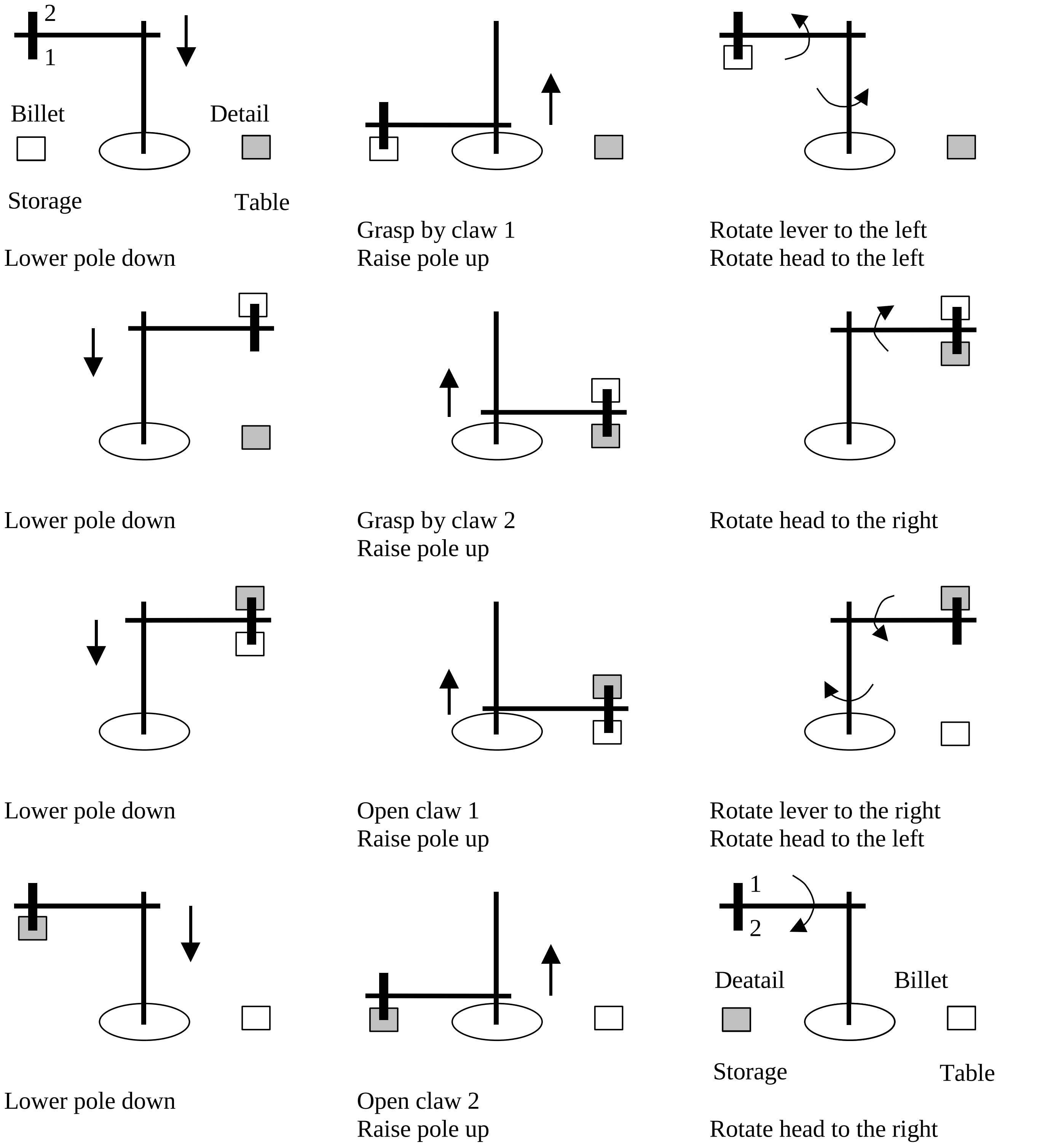
The significant achievement of describing technique is the possibility for representation both the algorithm of control and model of controlling object by Petri net. This allows the complex debugging of system.

Let's construct the control system of robot-manipulator:



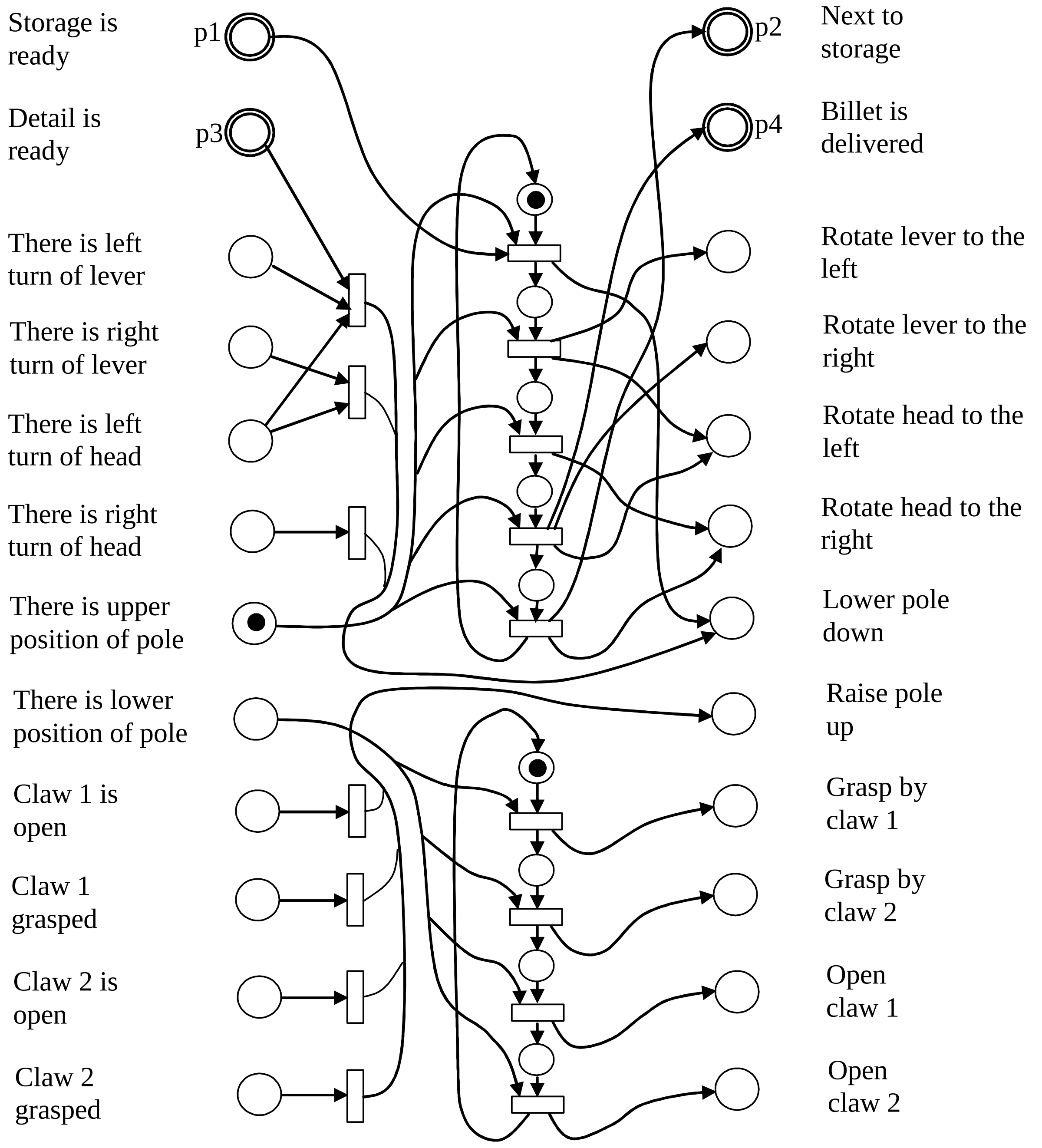
Robot interacts with storage of billets/details and rotary table on which the processing of billets is executed. Robot consists of pole, lever and head with two claws. Pole may rise up or lower down. There are two end sensors of upper and lover positions of the pole. Drive executes two commands: raise pole and lower pole. Lever rotates on 180 degrees and fixes two extreme positions over rotary table and over storage with the help of end sensors. Drive executes two commands: rotate left and rotate right. Head has two claws situates on opposite sides of it. It rotates on 180 degrees fixing in two extreme positions with the help of end sensors. Drive executes commands: rotate left and rotate right. Each of claws may either open or closed.

Initial position of parts is as follows: pole in the extreme upper position, lever in the extreme right position over storage, claw 1 directed down. Basic loop of robot's functioning consists in the following sequence of operations: lower pole down, grasp billet, raise pole up, rotate lever to the left and rotate head to the right, lower pole down, grasp detail by claw 2, raise pole up, rotate head to the right, lower pole down, open claw 1 for placement billet on rotary table, raise pole up, rotate lever to the right and rotate head to the left, lower pole down, open claw 2 for placement detail in storage, raise pole up, rotate head to the right. Let us represent described operations by the following sequence of pictograms:



Storage of billets/details constitutes rectangular box with cells for billets/details. Movements of storage in two directions provide the disposition of the required cell under the extreme right position of head. Rotary table provides the processing of detail in five positions and also an additional sixth position for loading/unloading under the extreme left position of head.

Let's represent the algorithm of robot's control:



Places p1, p2 are aimed to interaction with storage and places p3, p4 for interaction with rotary table. In the simplest case the model of the object may consists of a set of separate transitions representing the correct execution of command and firing of sensor. For instance, for pole it consists of transition connecting output place "lower pole down" with input place "there is lower position of pole".

It's required to input constructed algorithm into simulation system Tina (Appendix 3.2), debug standalone and together with the model of the controllable object.

Construct the models of control for storage and rotary table. Debug them standalone and together with the model of robot.

Investigate the properties of Petri net models constructed; prove the correctness of control algorithms with the help of system Tina (Appendix 3.2).

Variants of tasks

Variant	Size of storage		Size of rotary table
	Width	Length	
1	2	4	3
2	3	4	4
3	4	4	5
4	2	3	6
5	2	4	3
6	3	4	4
7	4	4	5
8	2	3	6
9	2	4	3
10	3	4	4
11	4	4	5
12	2	3	6
13	2	4	3
14	3	4	4
15	4	4	5
16	2	3	6

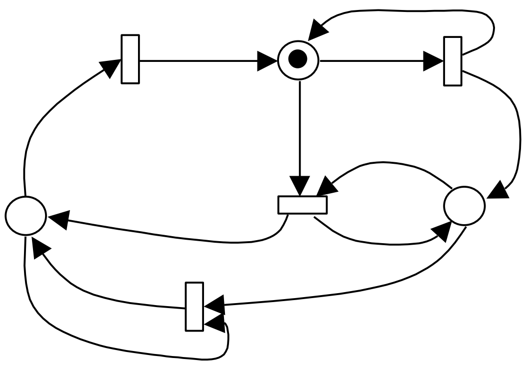
Test questions

1. What elements the standard scheme of control system consists of?
2. In what manner the interaction of controller and controllable object is implemented?
3. What the advantages of Petri nets application for realization of control systems consist in?
4. What properties have to possess a correct algorithm of control?

<u>Appendix 1</u>

Collection of Petri nets for investigation

1



2