



## Simple Switching Rules

In this section, the simple switching rules for SFC are explained. Later on when dealing with process synchronization, the switching rules will be investigated again. In order to setup the switching rules the following definition is necessary:

A transition is called ‚enabled‘ if all steps directly preceding the transition are marked. Otherwise the transition is called ‚disabled‘.

Firing a transition: If a token moves from one step to the next step, we say that the transition in-between the steps has been fired.

With this vocabulary, it is now easy to describe the switching rules: a transition can fire, when it is enabled and the transition condition is true. When the transition can be fired, the preceding step is executed once to set the action state according to their qualifiers. Then, the transition fires and the succeeding step become marked. If several transitions can fire, all will fire simultaneously. This completely defines the flow of tokens.

The processing with respect to the action is not so obvious. After firing a transition the crucial question is: Are the actions of the newly marked steps executed, if another transition can immediately fire? For the answer there exist two SFC variants. One variant can be named ‚Search for Stability‘ the other ‚No Search for Stability‘.

a) The ‚Search for Stability‘-Variant, which is implemented in many of the PLCs, the automata fires all transitions in the same iteration cycle until a stable marking is reached. A stable marking is defined as a state of the SFC, where the transition conditions of all enabled transitions are false. The flowchart for ‚Search of Stability‘ is shown in the next table on the right. The loop of firing transitions and calculating new marking is only left, if no transition can fire at that moment, i.e. a stable marking is obtained. Afterwards the set of active actions is determined and executed.

b) The flowchart for SFC-calculation for the ‚No Search for Stability‘-Variant is shown in the next table on the right. After firing all possible of the actual iteration cycle, the active actions are determined and executed and the iteration cycle is left.

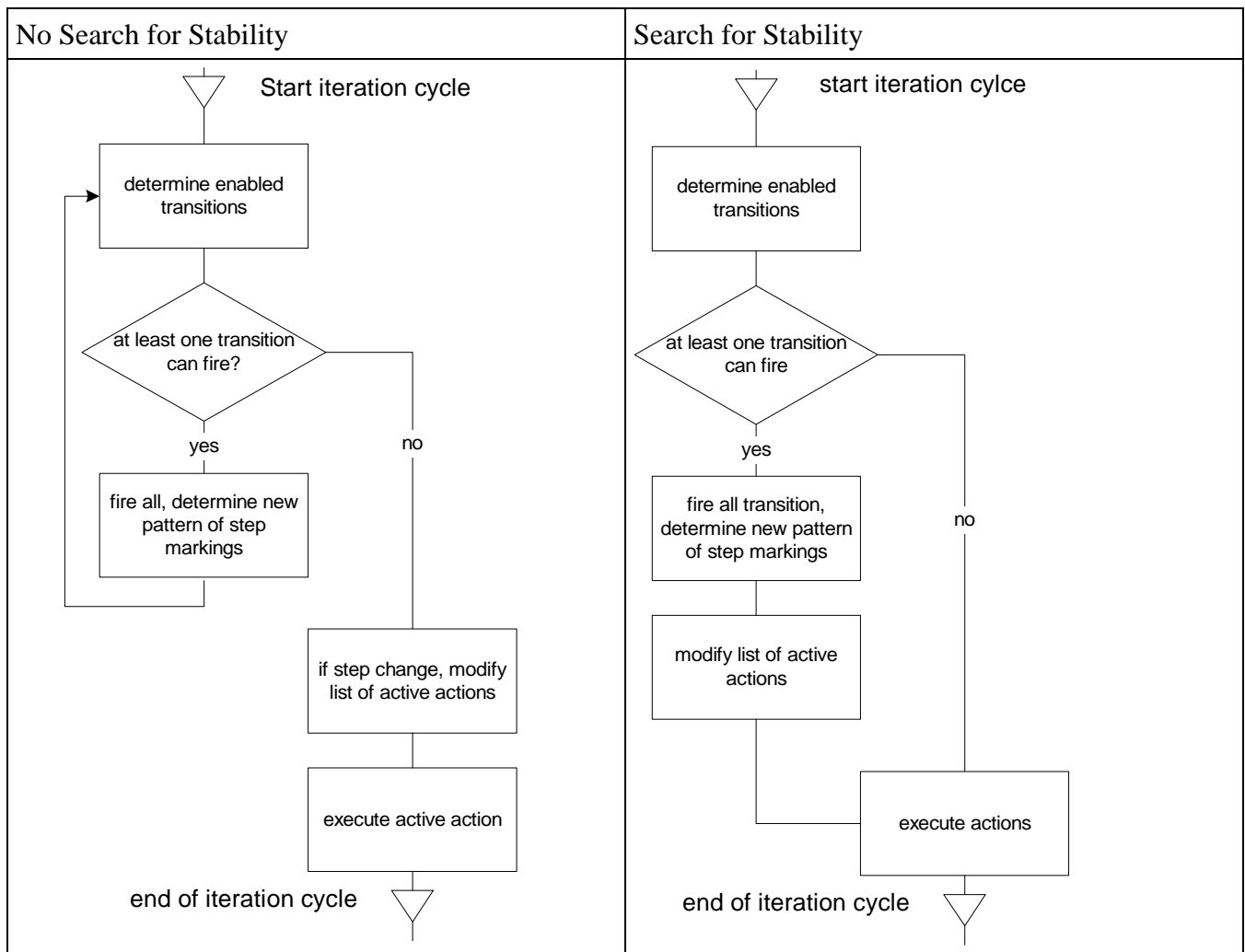
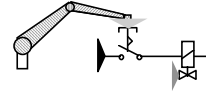


Table 1: Variants for SFC iteration

The consequence on action execution is shown with the previous simple example. Consider the case, when 'Motor Off' is marked (the motor is off) and the ON-Button is pressed but the 'Temperature high'-Condition is true. For the 'No Search for Stability' case, the SFC fires the ON-transition and step 'Motor ON' is marked. As a consequence, its actions are executed so the motor will be turned on. In the next iteration cycle the 'Temperature high'-transition is checked and the Alarm-step gets marked. The motor is turned off, because the action 'Motor Coil' isn't anymore active. In the 'Search for Stability'-case, the ON-transition fires, the step 'Motor On' is marked. Since the SFC checks now all the enabled transition, the 'Temperature high' is fired and step 'Alarm' is marked. Hopefully the 'Reset'-condition is not true and the SFC has reached a stable marking. Now the active actions are determined. The alarm lamp goes on and the action 'Motor Coil' was never executed. Obviously, the 'Search for Stability'-variant has the advantage, that the programmer doesn't need to check all failure conditions ahead of the on-command in order to prevent activation of undesired actions. On the other hand, it is easily possible to create an infinite loop, where no stable marking exists. Most PLC does not handle this case with the necessary precaution.

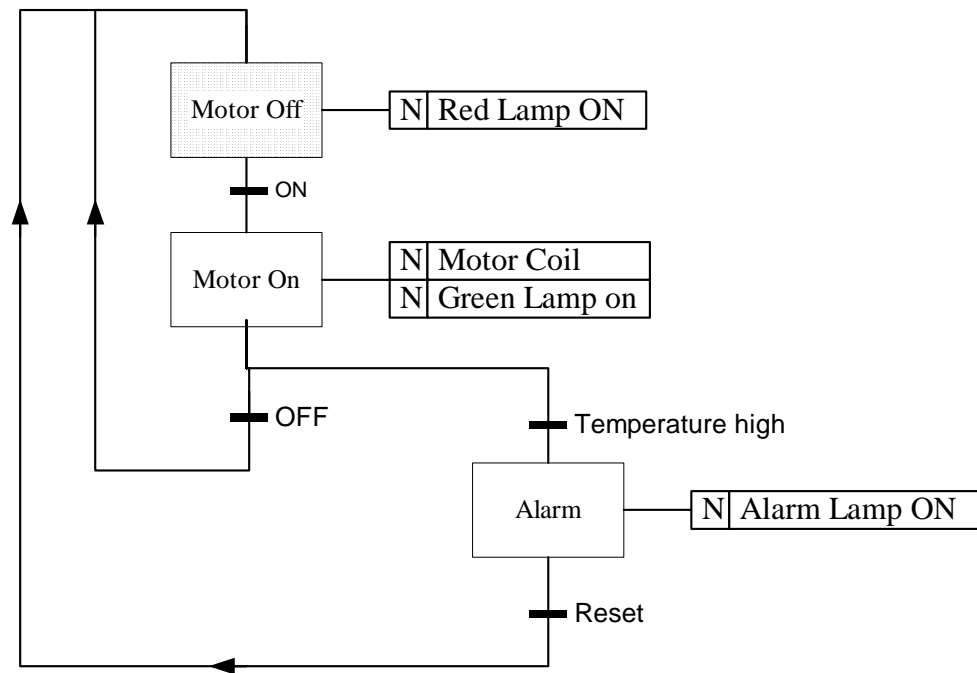
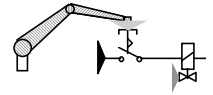


Figure 1: Example for action execution