### 16.1.4 Invoking functions from specific places

If value returning functions are called in receiving communication operations (in templates, template fields, in-line templates, or as actual parameters), in guards or events of alt statements or altsteps (see clause 20.2), or in initializations of altstep local definitions (see clause 16.2), the following operations shall not be used in functions called in the cases specified above, in order to avoid side effects that cause changing the state of the component or the actual snapshot and to prevent different results of subsequent evaluations on an unchanged snapshot:

1. All component operations, i.e. **create**, **start**(component), **stop**(component), **kill**, **running**(component), **alive,** **done** and **killed** (see notes 1, 3, 4 and 6).
2. All port operations, i.e. **start**(port), **stop**(port), **halt**, **clear**, **checkstate**, **send**, **receive**, **trigger**, **call**, **getcall**, **reply**, **getreply**, **raise**, **catch**, **check**, **connect**, **disconnect**, **map**, **unmap** (see notes 1, 2, 3, 4 and 6).
3. The **action** operation (see notes 2 and 6).
4. All timer operations, i.e. **start**(timer), **stop**(timer), **running**(timer), **read**, **timeout** (see notes 4 and 6).
5. Calling non-deterministic external functions, i.e. external functions where the resulting values for actual inout or out parameters or the return value may differ for different invocations with the same actual in and inout parameters (see notes 4 and 6).
6. Calling the **rnd** predefined function (see notes 4 and 6).
7. Changing of component variables, i.e. using component variables on the left-hand side of assignments, and in the instantiation of **out** and **inout** parameters (see notes 4 and 6).
8. Calling the **setverdict** operation (see notes 4 and 6).
9. Activation and deactivation of defaults, i.e. the **activate** and **deactivate** statements (see notes 5 and 6).
10. Calling functions and deterministic external functions with **out** or **inout** parameters (see notes 7 and 8).

NOTE 1: The execution of the operations **start**, **stop**, **done**, **killed**, **halt**, **clear**, **receive**, **trigger**, **getcall**, **getreply**, **catch** and **check** can cause changes to the current snapshot.

NOTE 2: The use of operations **send**, **call**, **reply**, **raise**, and **action** causes an error, i.e. all communication are to be made explicit and not as a side effect of another communication operation or the evaluation of a snapshot.

NOTE 3: The use of operations **map**, **unmap**, **connect**, **disconnect**, **create** causes an error, i.e. all configuration operations are to be made explicit, and not as a side effect of a communication operation or the evaluation of a snapshot.

NOTE 4: Calling of non-deterministic external functions, **rnd**, **running**, **alive**, **read**, **checkstate**, **setverdict**, and writing to component variables causes an error because this may lead to different results of subsequent evaluations of the same snapshot, thus, e.g. rendering deadlock detection impossible.

NOTE 5: The use of operations **activate** and **deactivate** causes an error because they modify the set of defaults that is considered during the evaluation of the current snapshot.

NOTE 6: Restrictions except the limitation on the use of **out** or **inout** parameterization in restriction j) apply recursively, i.e. it is disallowed to use them directly, or via an arbitrary long chain of function invocations.

NOTE 7: The restriction of calling functions and deterministic external functions with **out** or **inout** parameters does not apply recursively, i.e. calling functions that themselves call functions with **out** or **inout** parameters is legal.

NOTE 8: Using **out** or **inout** parameters causes an error because this may lead to different results of subsequent evaluations of the same snapshot.