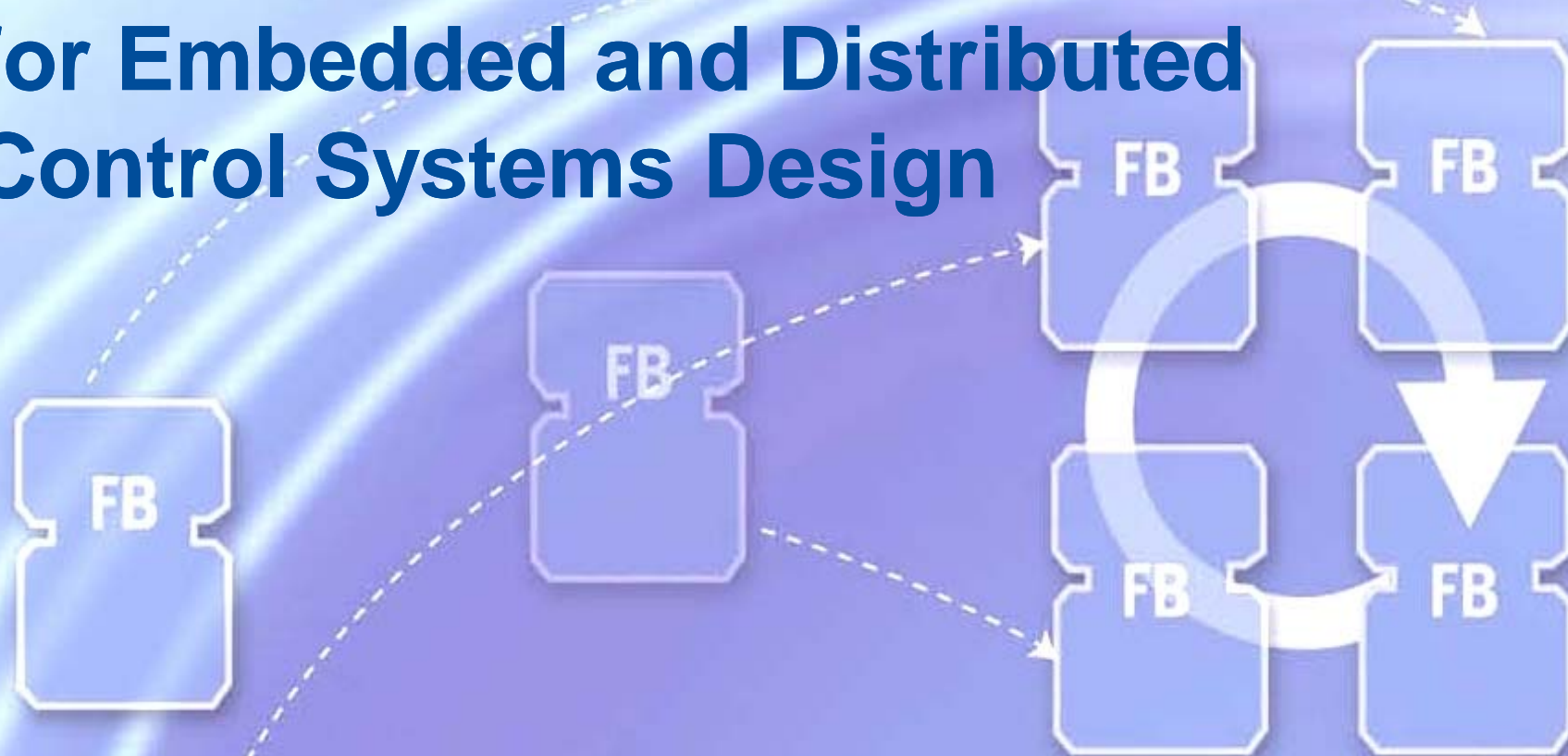


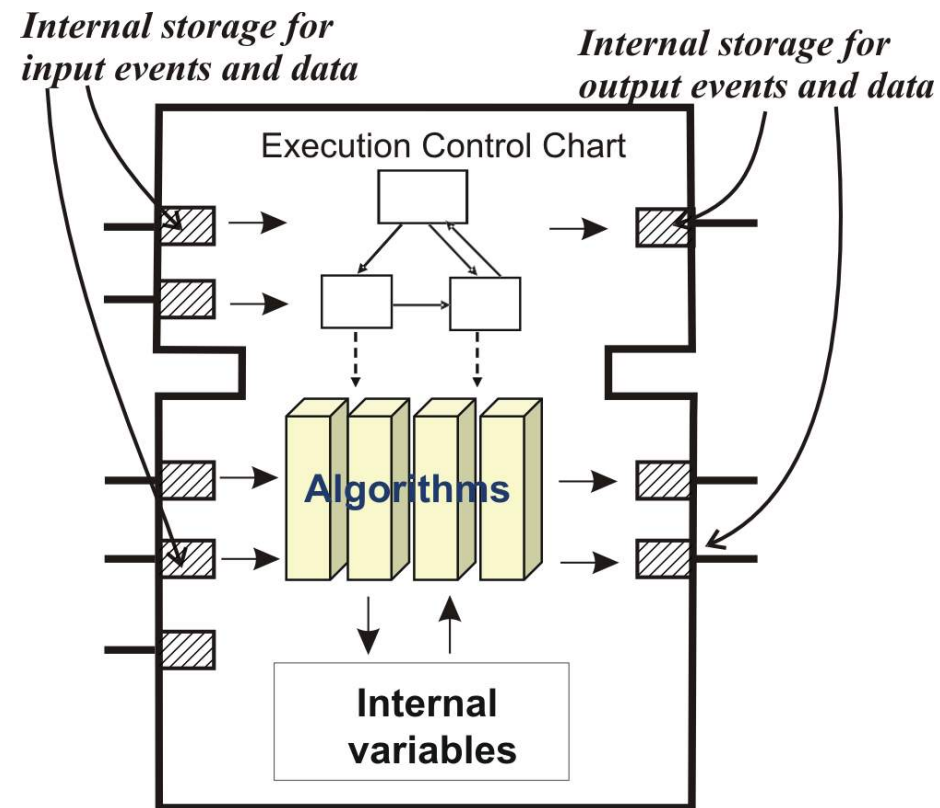
IEC 61499 Function Blocks for Embedded and Distributed Control Systems Design



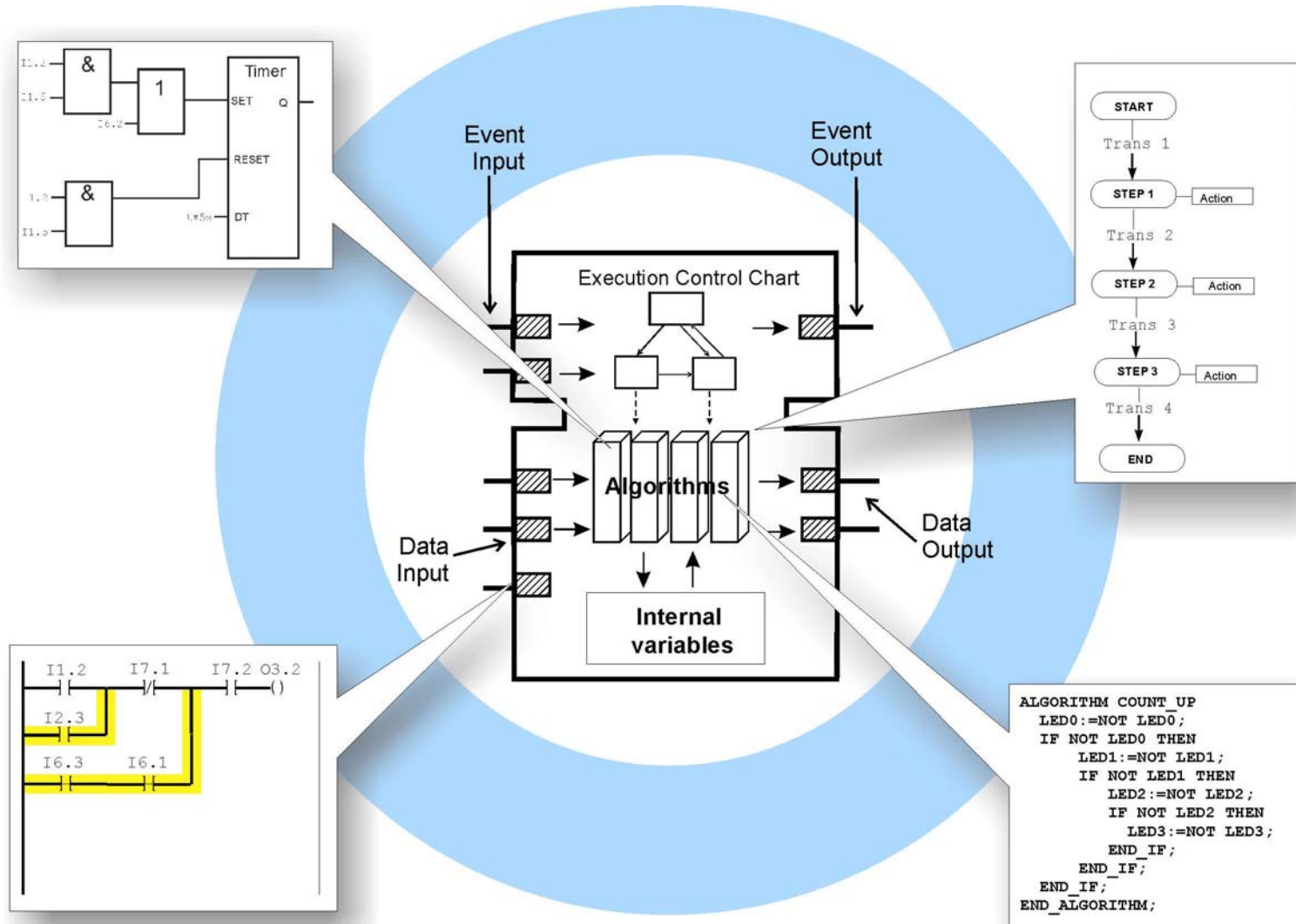
Lecture 7: Basic Function Blocks

Valeriy Vyatkin © 2007

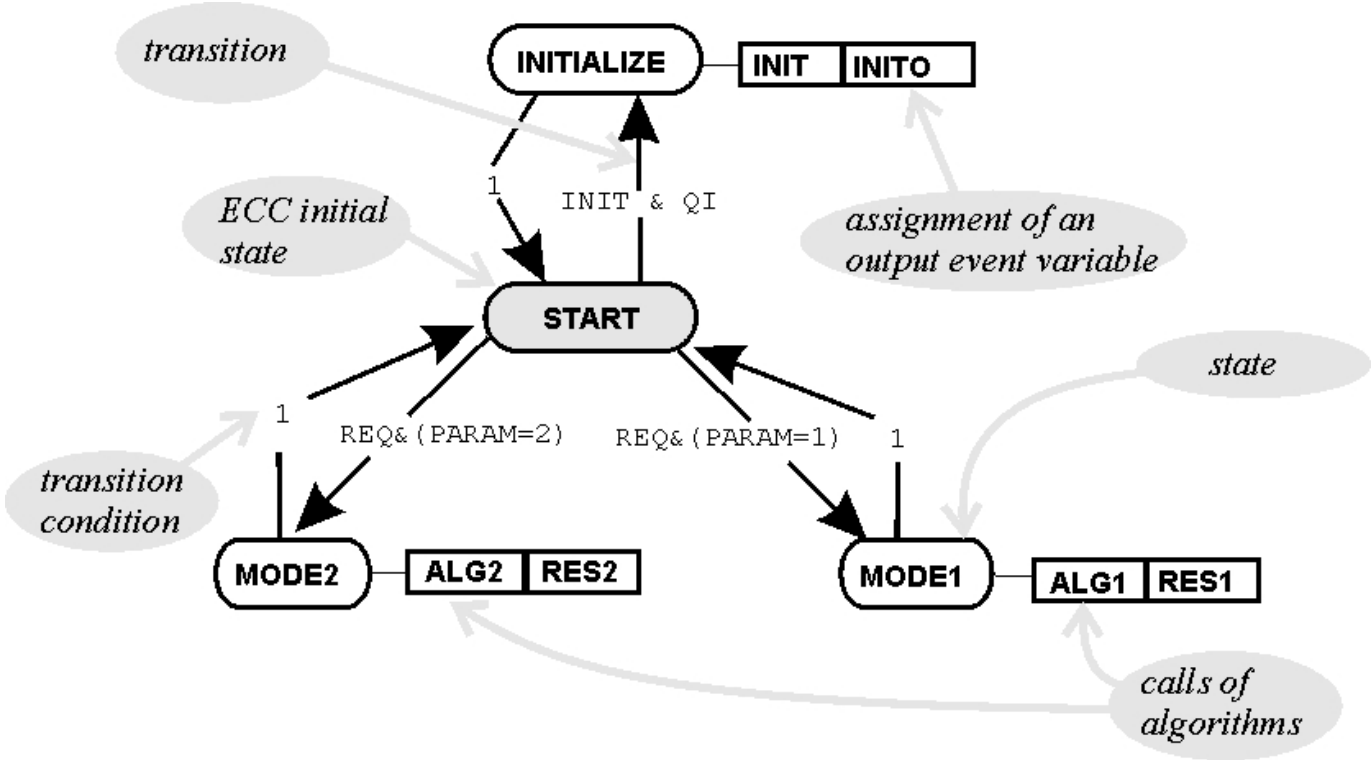
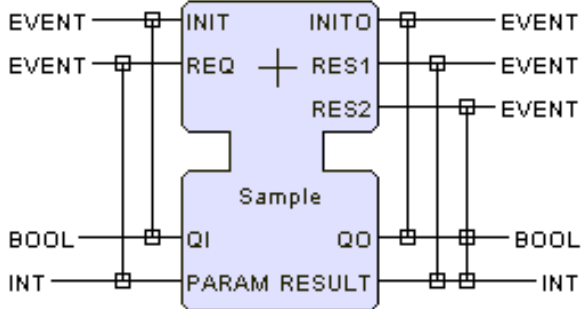
Basic Function Block



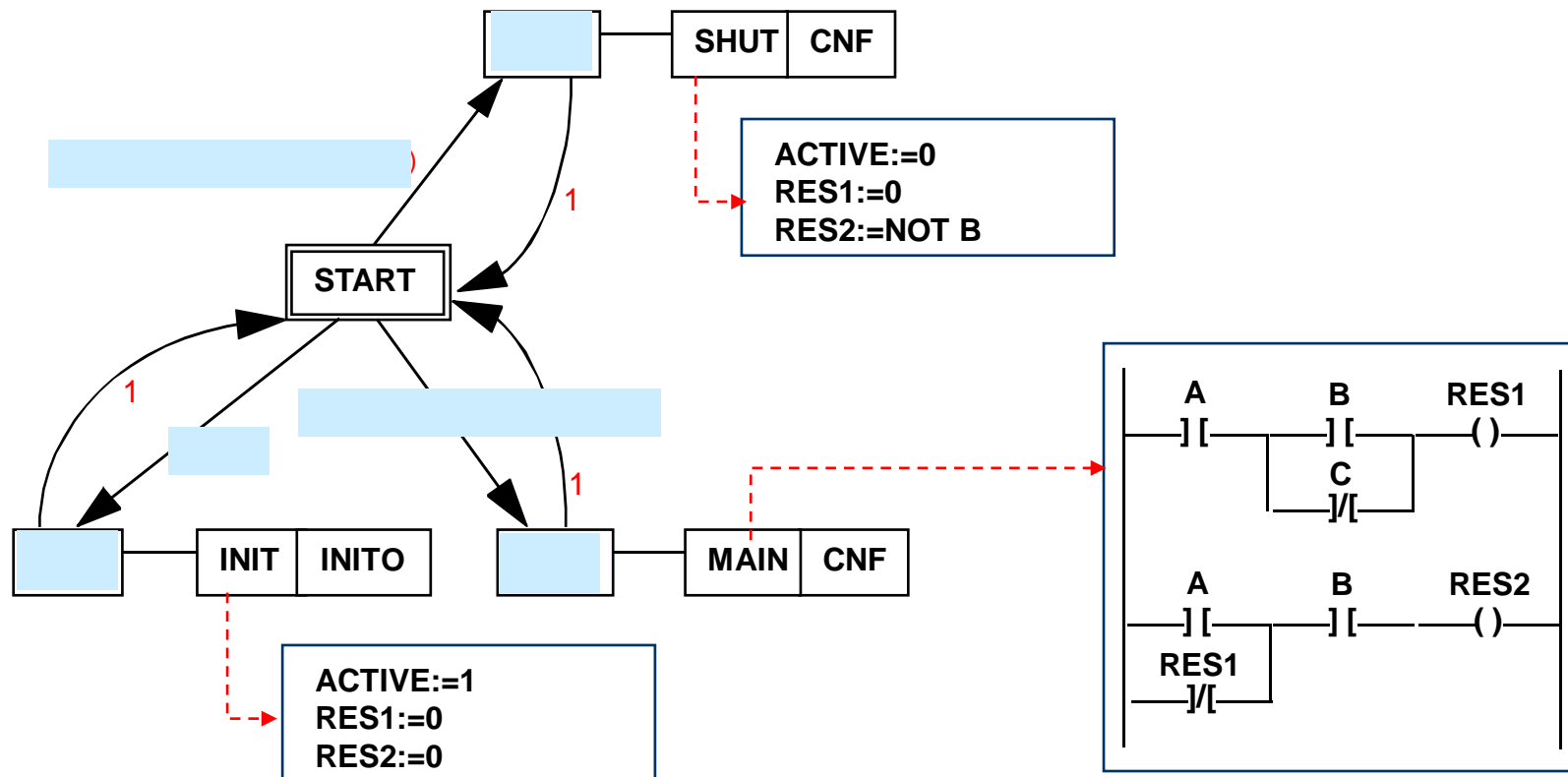
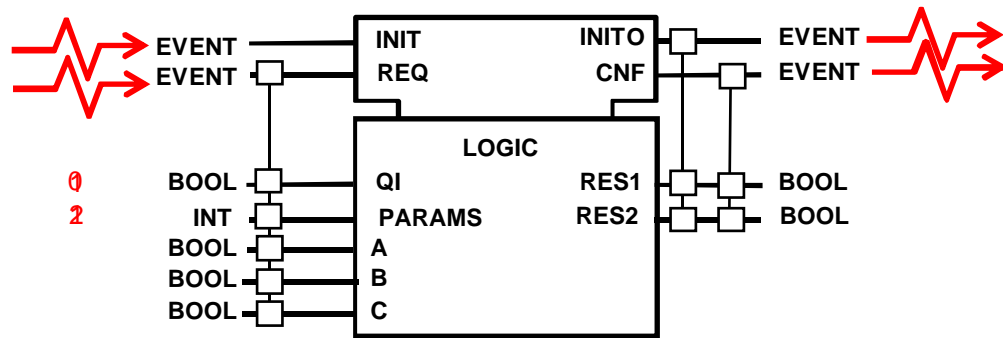
Encapsulation of legacy code



Execution Control Chart



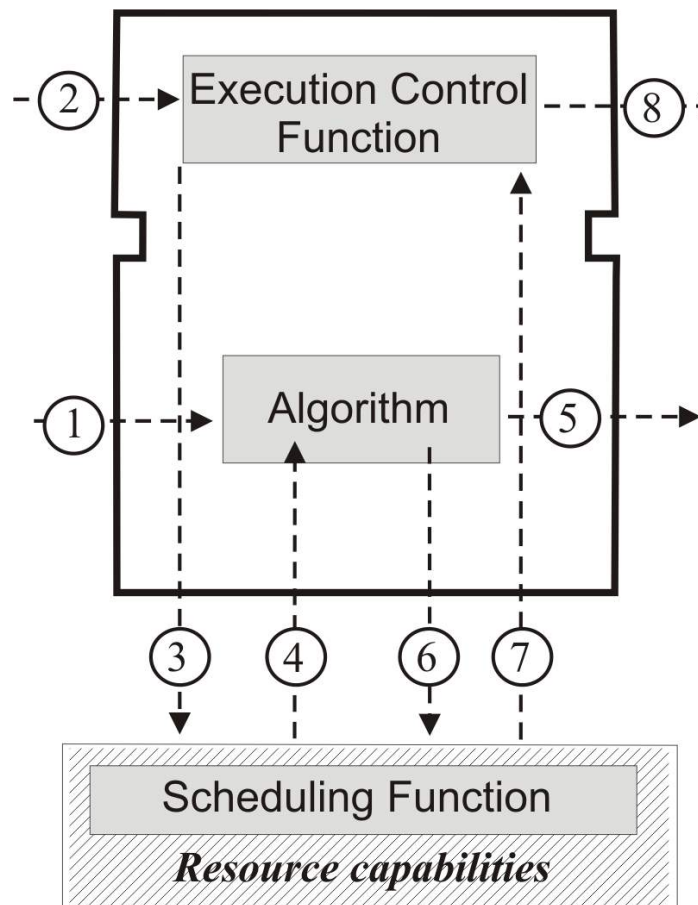
Execution model – event driven



Execution model

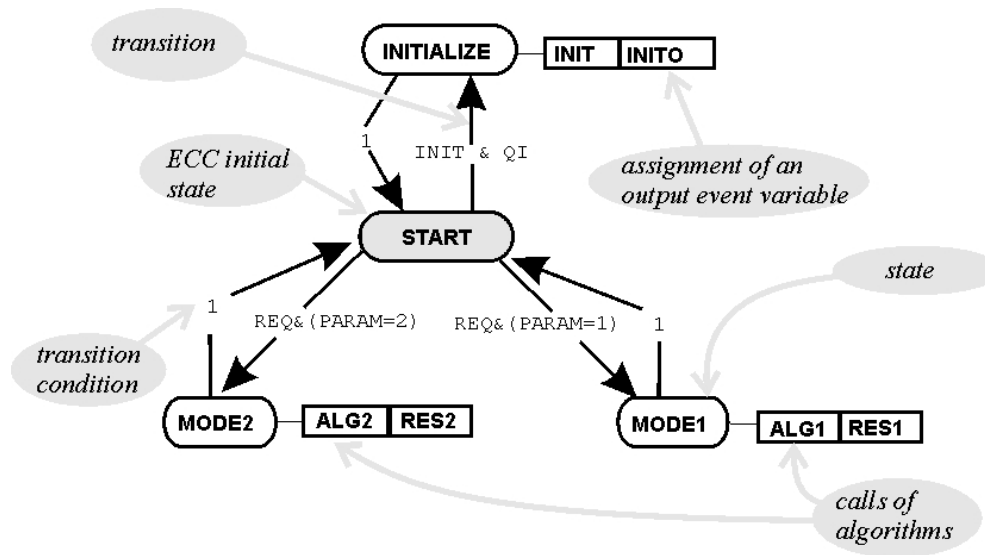
- **Syntax of transition conditions:**
 - **Event | Expression over Data | Event & Expression over Data**
- **No flip-flop memory for input event variables is required**
- **This simplifies the ECC interpretation rules making ECC very similar to Harel's State Charts**

Execution Sequence



- **Step 1:** The input variable values relevant to the input event are made available.
- **Step 2:** The input event occurs, the execution control of the function block is triggered
- **Step 3:** The execution control function evaluates the ECC and notifies the scheduling function to schedule algorithm for execution
- **Step 4:** Algorithm execution begins.
- **Step 5:** The algorithm completes the establishment of values for the output variables associated with the event output by the WITH qualifier
- **Step 6:** The resource scheduling function is notified that algorithm execution has ended.
- **Step 7:** The scheduling function invokes the execution control function.
- **Step 8:** The execution control function signals event at the event output.

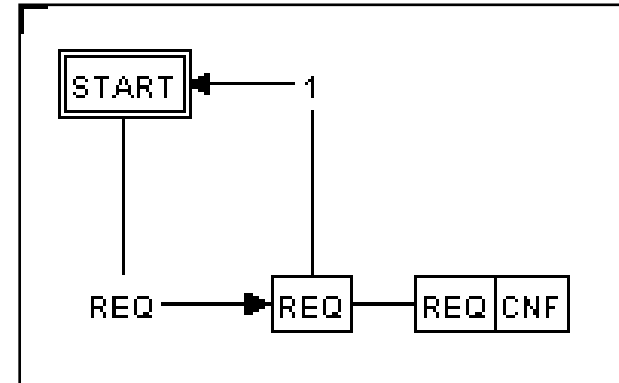
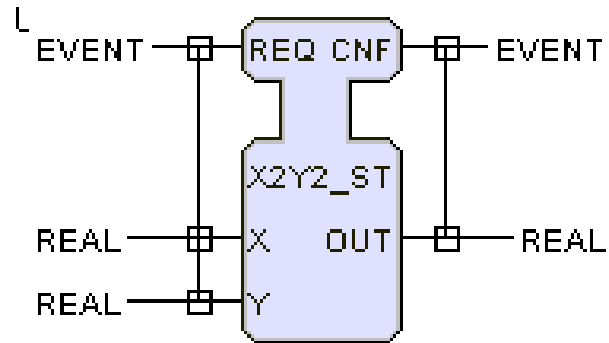
ECC interpretation example



1. All the transition conditions going out of the current ECC state are evaluated.
2. If no transition is enabled, then the procedure ends.
3. If one or several state transitions are enabled (i.e. if the corresponding conditions evaluated to TRUE), a single state transition takes place*.
4. The current state is substituted by the following one.
5. The algorithms associated with the new current state will be scheduled for execution.
6. The execution control function notifies the resource scheduling function to schedule an algorithm for execution.

* The order in which the transition conditions are evaluated corresponds to the order in which they are declared following the textual syntax

Example



ALGORITHM REQ IN ST
OUT := (X-Y)*(X+Y);
END

Summary

- A Basic Function Block is a platform-independent abstraction of a software component that is adjusted for applications in measurement and control systems.
- **The standard implies separation of the functions, implemented by algorithms, from the execution control.** The algorithms encapsulated in a function block can be programmed in different programming languages.
- **The execution of function blocks is event-driven.** This means that algorithms are executed only if an input event activates the block in contrast to the cyclically scanned execution of programs in IEC 61131.

Summary (continued)

- **The source of events can be other function blocks.** Some of them may encapsulate interfaces to the environment (controlled process, communication networks, hardware of a particular computational device).
- **The execution function of a Basic Function Block is defined in a form of a state machine** that is available for documentation and specification purposes even if the source code of algorithms is hidden.
- **The function block abstracts from a physical platform (the resource) where it is located on.** This means that the specification of the function block can be done without any knowledge of the particular hardware on which it will be later executed.