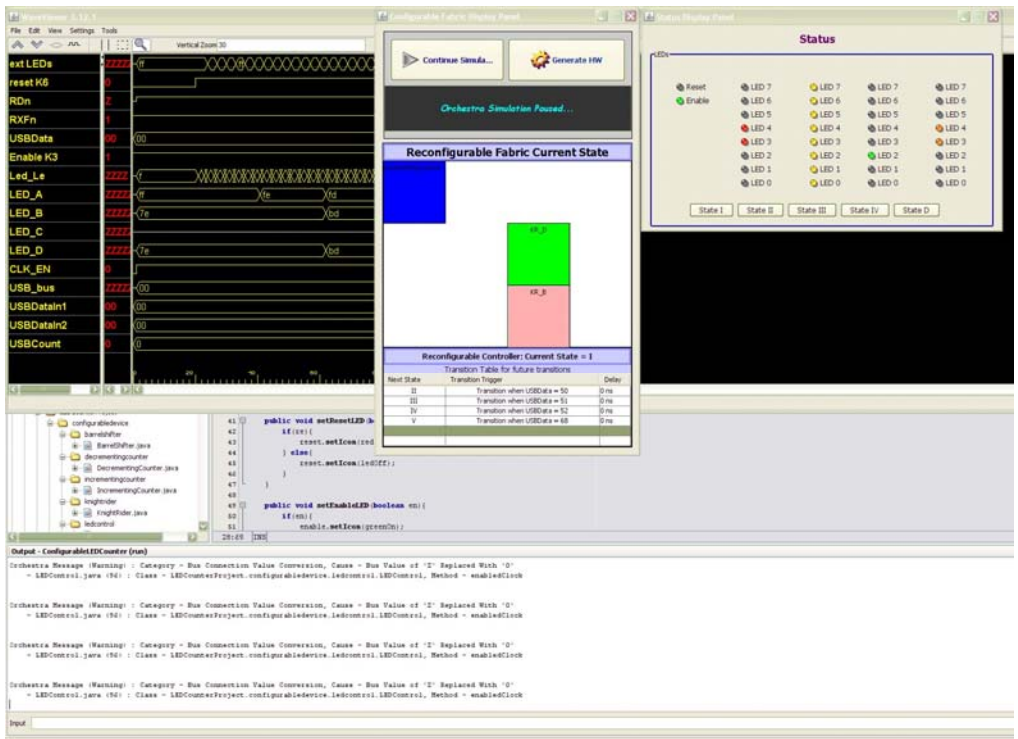
# High-level Modeling/Simulation in Orchestra

- Design is done at higher levels of abstraction (Java and OHDL)
- Orchestra environment provides for cycle-accurate simulations
  - Enhanced visibility and debugging tools
  - Allows you to watch the condition of the configurable logic and see state transitions
- Fast, event-driven simulation engine
  - Quick debug cycles
- Significantly reduces the development time and cost



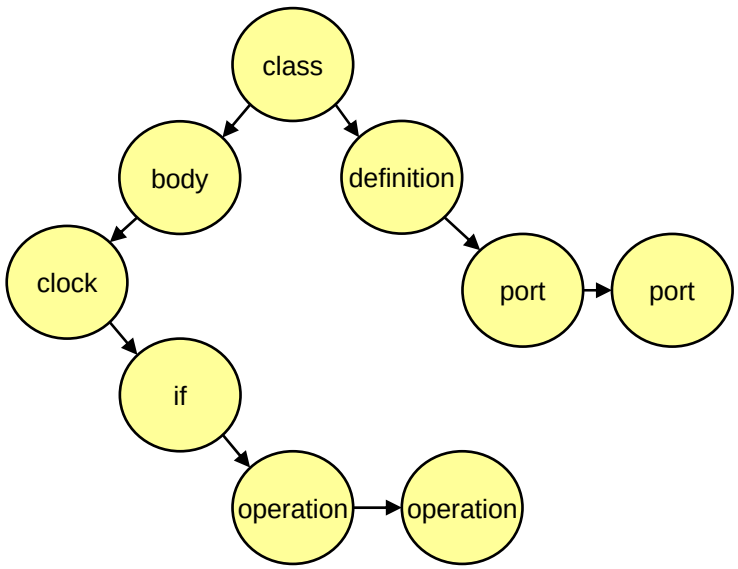



Sandia National Laboratories

# OHDL to VHDL Translation

- **Modules within the reconfigurable fabric are translated to VHDL from OHDL**
  - **File is parsed into a tree structure containing all the necessary information**
  - **Tree structure is run through a VHDL generation process to generate the file**
- **OHDL and VHDL modules are behaviorally identical**
- **Hooks included to allow for translation to any other HDL or ML**

File.java




File.vhd

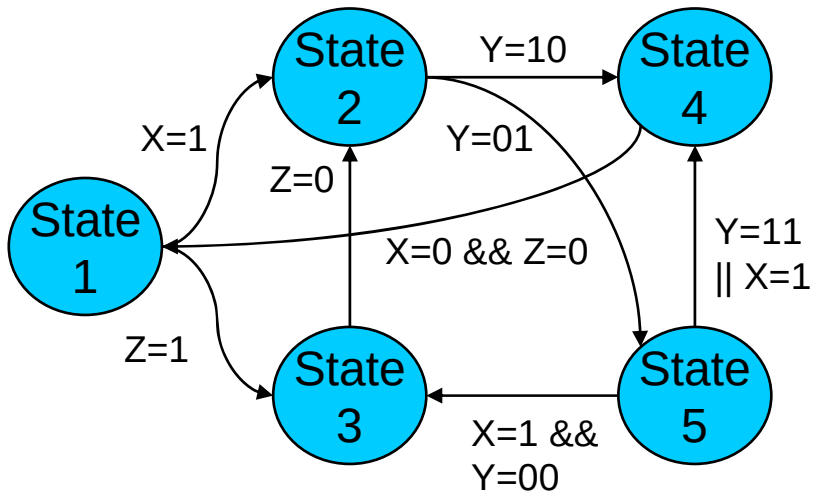


Sandia National Laboratories

# Configuration Controller Generation

- In software, the state table must be defined
  - Determine which modules exist in which states
  - Create a unique identifier for each state as well as a priority
- Also in software, the transition table must be defined
  - What signals cause transitions from each state
  - What state to enter due to each transition
- These tables define the configuration controller and allow simulation of partial reconfiguration
- Hardware behavior during reconfiguration is user-controlled
- The tools will generate a hardware-equivalent of the configuration controller in VHDL

| | Hardware Module | | | |
|---|---|---|---|---|
| State | A | B | C | D |
| 1 | 0 | 0 | 1 | 1 |
| 2 | 0 | 1 | 1 | 0 |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 1 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 0 |

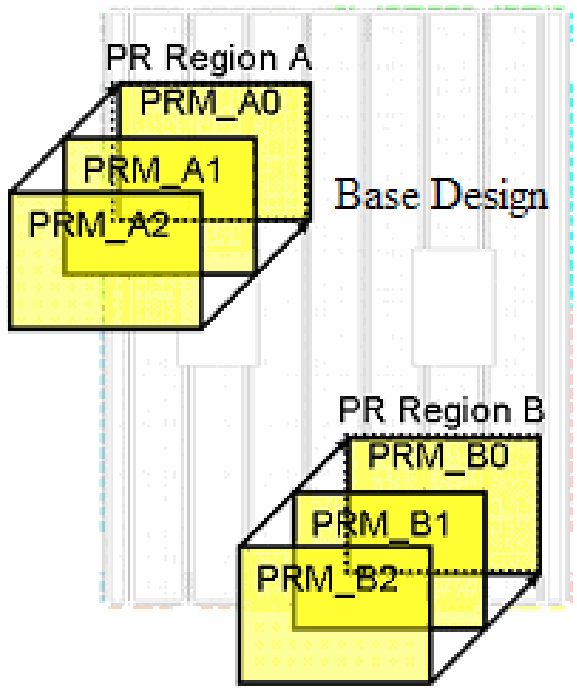





Sandia National Laboratories

# Region Partitioning/Floorplanning

- **Knowing the state table, the tools optimally allocate sets of hardware modules to share distinct regions**
  - **Requirements:**
    - **Modules that share a region must not exist at the same time in the state table**
    - **Modules that share an output port must exist in the same region**
- **Once the regions have been partitioned, they must be physically constrained onto the chip**
  - **Define the size, shape, and placement of the region**
    - **Must be large enough to accommodate the biggest module**
    - **Must be shaped and placed to allow the design to meet timing**
  - **Must obey internal FPGA placement restrictions**
    - **Frame Boundaries**
    - **Embedded I/O**
    - **RAMB16/DSP48 Column Boundaries**
  - **Must also locate and constrain bus macros**






Sandia National Laboratories

# Partial Bitstream Generation

- Vendor/FPGA specific step
- Special PR Service Pack
  - Early access flow not available to the general public at this time
- Vendor design flow accessed through generated script files
- The tools generate:
  - Partial bitstreams for each reconfigurable module
  - Total bitstreams for each individual state
  - Blanking bitstreams for each reconfigurable region
- Timing is checked for each individual state
- Bitstreams are parsed and downloaded to a bank of non-volatile flash memories



Sandia National Laboratories