

## **HDL – Hardware Description Language**

### **Two principle HDLs currently used:**

- VHDL – VHSIC Hardware Description Language  
VHSIC – Very High Speed Integrated Circuit  
gate level through system level design and verification
- Verilog is other main HDL  
Primarily targeted for design of ASICs  
ASIC – Application Specific Integrated Circuit

**Synthesis** –conversion of an HDL description to gate level design  
*Current HDLs owe their success to synthesis tools!!!*

### **Benefits of HDLs:**

- Early design verification via high level design verification
- Evaluation of alternative architectures
- Top-down design (w/synthesis)
- Reduced risk to project due to design errors
- Design capture (w/synthesis & independence of implementation media)
- Reduced design/development time & cost (w/synthesis)
- Base line testing of lower level design representations - example: gate level or register level design
- Ability to manage/develop complex designs
- Hardware/software co-design
- Documentation of design (*depends on quality of designer comments*)

### **Designer concerns about HDLs:**

- Loss of control of detailed design
- Synthesis is inefficient
- Quality of synthesis varies between synthesis tools
- Synthesized logic does not perform the same as the HDL
- Learning curve associated with HDLs & synthesis tools

**Basic Domains of HDLs:**

- Structural – components and their interconnections (netlist)
- Behavioral – describes I/O responses & behavior of design
- Register Transfer Level (RTL) – a data flow description at the register level

**Design space issues (may be critical to project or traded-off):**

- Area (chip area, how many chips, how much board space)
- Speed/performance
- Cost of product
- Production volume
- Design time (to meet market window & development cost)
- Risk to project (working, cost-effective product on schedule)
- Reusable resources (same circuit - different modes of operation)
- Implementation media
- Technology limits
- Designer experience
- CAD tool availability and capabilities

**Requirements DoD placed on VHDL in mid 80s:**

- An HDL for:
  1. Design & description of hardware
  2. Simulation & documentation (*with designer comments*)
  3. Design verification & testing
- Concurrency to accurately reflect behavior & operation of hardware— all hardware operates concurrently (*as a result, all VHDL simulation is event-driven*)
- Hierarchical design – essential for efficient, low-risk design
- Library support – for reuse and previously verified components
- Generic design
  1. independent of implementation media
  2. can be optimized for area or performance
- Timing control – to assign delays for more accurate simulation
- Portability between simulators & synthesis tools (*not always true*)

## Typical Product Development & Design Verification Cycle Using HDLs

