

Term Project Milestone 4 Evaluation (Components Specification)

Evaluation Criteria Categories	Specific Criteria	Comments	Score
Consistency with higher level specifications	<ul style="list-style-type: none"> <input type="checkbox"/> Components have interfaces (inputs, outputs, and control signals) that are consistent with the datapath specification, including signal widths. <input type="checkbox"/> Components produce behaviors that are consistent with the assembly language and register transfer language levels of the design specification. <input type="checkbox"/> Components implement their behaviors within the timing constraints imposed by the RTL specification. 		(3)
Self-consistency	<ul style="list-style-type: none"> <input type="checkbox"/> Example: Specification of 1-bit ALU is consistent with specification of 16-bit ALU. <input type="checkbox"/> Example: Specification of bi-directional variable-displacement shifter is consistent with specifications of unidirectional variable-displacement shifters. 		(3)
Demonstration of design principles 1. Simplicity favors regularity 2. Smaller is faster 3. Good design demands good compromises 4. Make the common case fast	<ul style="list-style-type: none"> <input type="checkbox"/> Component specifications are as simple as reasonably possible (e.g. variable-displacement shifters are composed of multiple fixed-displacement shifters). <input type="checkbox"/> Component specifications are as small as reasonably possible (e.g. variable-displacement shifters use as few fixed-displacement shifters as possible). <input type="checkbox"/> Conflicts between the preceding criteria are resolved by considering overall performance (e.g. design of variable displacement shifters considers how often shifts of various displacements actually are used) 		(3)
Documentation (see below) <input type="checkbox"/> Organization <input type="checkbox"/> Completeness <input type="checkbox"/> Conciseness <input type="checkbox"/> Grammar and style	<ul style="list-style-type: none"> <input type="checkbox"/> All design decisions necessary to implement Xilinx model are documented (components may be implemented by core generated objects or built-in symbols, which include gates and some higher-level entities) <input type="checkbox"/> Clear English specifications as necessary <input type="checkbox"/> Component tests 		(16)

Required Documents

- Memo
 - Objective assessment of design and status
- Design Documentation
 - Demonstration of conceptual understanding
 - Highlights interesting features
- Design Process Journal
 - Alternatives considered
 - Tradeoffs
 - Decisions
- Website