

Table 2.2
Specification of ALU instruction type

| Generic Subcomputations | ALU Instruction Type | |
|-------------------------|-------------------------------------|-------------------------------------|
| | Integer Instruction | Floating-Point Instruction |
| IF | Fetch instruction (access I-cache). | Fetch instruction (access I-cache). |
| ID | Decode instruction. | Decode instruction. |
| OF | Access register file. | Access FP register file. |
| EX | Perform ALU operation. | Perform FP operation. |
| OS | Write back to register file. | Write back to FP register file. |

Table 2.3
Specification of load/store instruction type

| Generic Subcomputations | Load/Store Instruction Type | |
|-------------------------|--|---|
| | Load Instruction | Store Instruction |
| IF | Fetch instruction (access I-cache). | Fetch instruction (access I-cache). |
| ID | Decode instruction. | Decode instruction. |
| OF | Access register file (base address). Generate effective address (base + offset). Access (read) memory location (access D-cache). | Access register file (register operand, and base address). |
| EX | | |
| OS | Write back to register file. | Generate effective address (base + offset). Access (write) memory location (access D-cache). |

Table 2.4
Specification of branch instruction type

| Generic Subcomputations | Branch Instruction Type | |
|-------------------------|---|---|
| | Jump (unconditional) Instruction | Conditional Branch Instruction |
| IF | Fetch instruction (access I-cache). | Fetch instruction (access I-cache). |
| ID | Decode instruction. | Decode instruction. |
| OF | Access register file (base address). Generate effective address (base + offset). | Access register file (base address). Generate effective address (base + offset). |
| EX | | Evaluate branch condition. |
| OS | Update program counter with target address. | If condition is true, update program counter |