CDA 4205 Computer Architecture

Processor Components Implementation Part I

1. **Objectives**

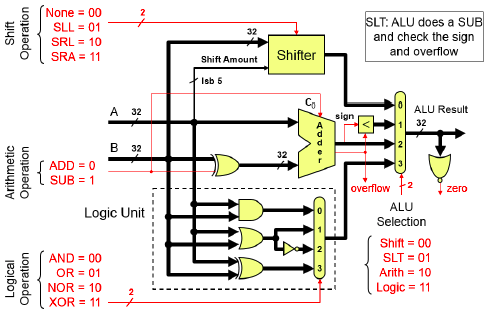
Using the Logisim Simulator to design the main data path components: a 32 bit arithmetic and logic unit (ALU).

1. **Tasks**

Design a 32-bit ALU to perform all the arithmetic, logic and shift operations required by your data path, and model the your designed 32-bit ALU in Logisim, then test the correct functionality of all operations implemented by the ALU.

1. **Components Design Specifications**
2. **Arithmetic and Logic Unit (ALU) Design**

The designed ALU should have four main units as shown in the following picture: arithmetic unit, comparator unit, logic operation unit and shifter unit to perform arithmetic, comparison, and logic and shift operations, respectively. In order to select the output from either the arithmetic unit, the comparator unit, the logical operation unit for the shifter unit, a 32-bit 4 x 1 multiplexer is used with 2-bit ALU selection signals.



* Arithmetic Unit

1. Input signals: two input 32-bit integers A and B, one control signal ADD/SUB.
2. Output signals: A+B or A-B, Carry-Out (Cout) and Overflow signal
3. Building Components: 32-bit full adder, 32-XOR gates

The arithmetic Unit is composed of a 32-bit adder that can perform 32-bit addition and subtraction. Its internal implementation can be designed using a 32 ripple carry full adders. The inputs A and B are two 32-bit integers and the output F is A+B or A-B. The arithmetic block has a control signal, ADD/SUB, to determine whether the operation to be performed is addition or subtraction. If this signal is 0, the adder will perform addition, otherwise it will perform subtraction. The subtraction is performed using 2's complement representation as A – B = A + B' + 1. B' is computed by the 32-XOR gates in the arithmetic unit. The adder also generates Carry-Out (Cout) and Overflow signal that can be used to test for comparison purposes for unsigned and signed operations and for correctness of the obtained result. The overflow signal can be generated by XORing the carry-outs of bits 30 and 31.

* Comparator Unit

The comparator unit is mainly used for comparing signed and unsigned numbers and it shares the 32-bit full adder with the Arithmetic Unit. For the MIPS ISA implementation, we only need to compare if a number is less than another number for implementing the set on less than instructions: SLT, SLTU, and SLTI.

In order to compare signed numbers A and B, we perform the subtraction operation A – B and then we check both the Sign of the result and the Overflow signal. The Sign of the result is the most significant bit of the result. From the flowing table, we can see that if the Sign value is not equal to the Overflow value, then A < B, otherwise, A ≥ B. This can be done by XORing the Sign and the Overflow signals. If the result is 1, this means that A < B, otherwise, means A ≥ B.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sign** | **Overflow** | **Results** | **Comments** |
| 0 | 0 | A ≥ B | No overflow and positive difference. |
| 0 | 1 | A < B | A < 0, B > 0 and overflow |
| 1 | 0 | A < B | No overflow and negative difference. |
| 1 | 1 | A ≥ B | A > 0, B < 0 and overflow |

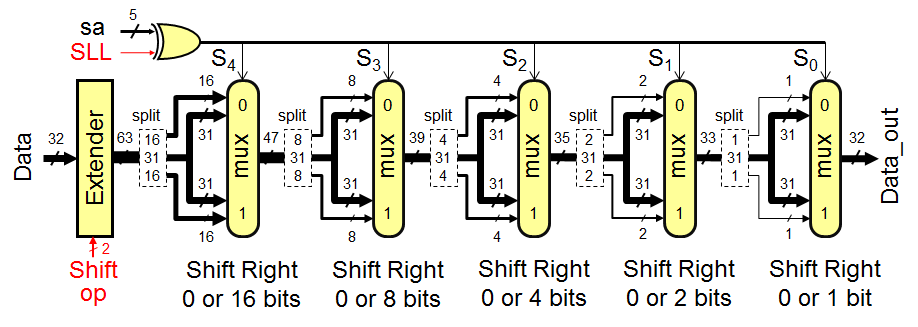
For unsigned comparison of two numbers A and B, we need to perform a subtraction operation, A - B, and then check whether we have a Carry-Out (Cout) or not. Note, we use signed number full-adder to perform unsigned number subtraction. If Carry-Out=0, this means that there was a borrow, thus A < B. However, if Carry-Out=1 then this implies that there was no borrow and hence A ≥ B.

* Logical Operation Unit

The designed logical operation unit should perform AND, OR, NOR, and XOR logical operations for MIPS logic instruction. Functions in this unit are performed using the basic logic gates. AND, OR, NOR and XOR logical gates outputs will be forwarded to the inputs to a 32-bit 4 x 1 multiplexer. The logical operation unit has two control bits to determine the logical operation is AND, OR, NOR, or XOR, which will be connected to the input selection bits of the multiplexer.

* Shifter Block

The shifter block is used to implement SLL (shift left logic), SRL (shift right logic), SRA (shift right arithmetic) and ROR (rotate right) MIPS shift instructions. A shift operation takes a binary number and shifts it left or right by a specified number of bits. One possible implementation of the shifter known as the Barrel Shifter is given below. This architecture has the advantage of performing a number of operations using the same hardware.



The input data is extended from 32 to 63 bits as follows:

* If shift op = SRL, then ext\_data[62:0] = 031 || data[31:0]
* If shift op = SRA, then ext\_data[62:0] = data[31]31 || data[31:0]
* If shift op = ROR, then ext\_data[62:0] = data[30:0] || data[31:0]
* If shift op = SLL, then ext\_data[62:0] = data[31:0] || 031

For SRL, the 32-bit input data is zero-extended to 63 bits. For SRA, the 32-bit input data is sign-extended to 63 bits. For ROR, 31-bit extension = lower 31 bits of data. Then, shift right according to the shift amount

* Zero Flag Detector Block

The role of this block is to set the zero-flag bit to 1 whenever the output of any operation is equal to zero. This could easily be designed using a NOR gate at the output.

1. **The report document must contain sections listed in the following:**
2. **Design and Implementation**

* Specify clearly the design giving detailed description of the designed ALU, its components, control, and the implementation details.
* Provide drawings of the component circuits.

1. **Simulation and Testing**

* Carry out the simulation developed using Logisim.
* Describe the testing procedure, its inputs, and its expected output.
* Provide snapshots of the Simulator window with your test input data and showing the simulation output results.
* **Submission Requirements**
* All submissions will be done through canvas.
* Attach one zip file containing all the design circuits and sub-circuits, as well as the report document.
* Only submissions via the link on Canvas where this description is downloaded are graded. Submissions to any other locations on Canvas will be ignored.
* Late submissions are accepted for a maximum of 3 late days with 20% assignment credit off as late penalization. Assignments submitted after 3 late days will not be accepted.