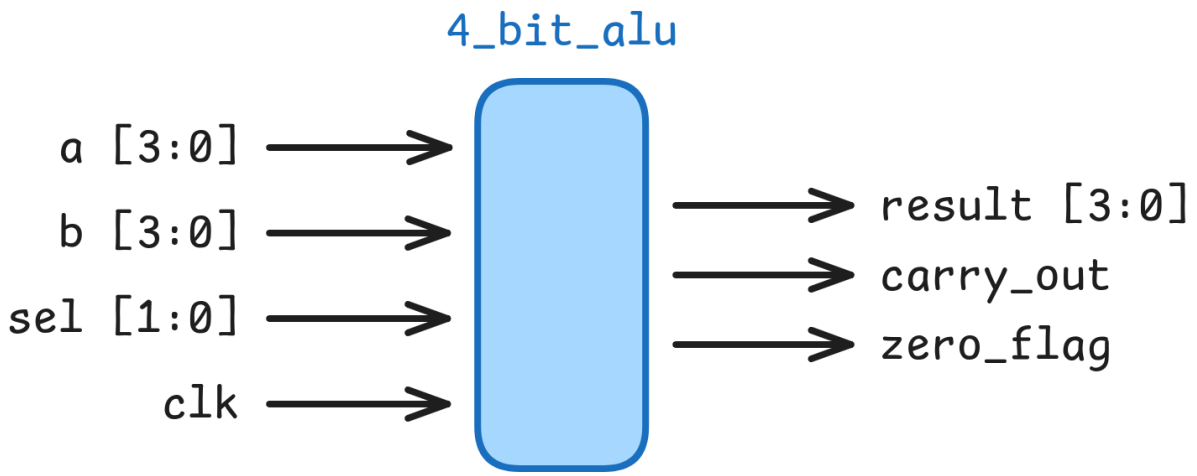
ALU 4 Bits

4 Bits ALU

- **Full Name:** Nelson Rodriguez
- **Project Name:** 4-Bit Synchronous ALU
- **Brief Description:** This module is a 4-bit synchronous Arithmetic Logic Unit (ALU). It takes two 4-bit operands (`a` and `b`) and a 2-bit selector signal (`sel`) to determine the operation: Addition, Subtraction, bitwise AND, or bitwise OR. The outputs are registered on every rising edge of the clock (`clk`). The module provides a 4-bit output (`result`), a `carry_out` flag for arithmetic operations, and a `zero_flag` that is asserted when the result is zero.

Block Diagram



ALU Functional Table

This table shows the expected outputs for sample inputs for each of the four operations selected by the `sel` signal.

`sel`	Operation	Example Inputs	Expected Outputs
		`a`, `b`	`result`, `carry_out`, `zero_flag`
`00`	ADD	`5` (`0101`), `3` (`0011`)	`8` (`1000`), `0`, `0`
`00`	ADD	`10` (`1010`), `7` (`0111`)	`1` (`0001`), `1`, `0`
`01`	SUB	`9` (`1001`), `4` (`0100`)	`5` (`0101`), `0`, `0`
`01`	SUB	`5` (`0101`), `5` (`0101`)	`0` (`0000`), `0`, `1`
`10`	AND	`12` (`1100`), `10` (`1010`)	`8` (`1000`), `0`, `0`

`sel`	Operation	Example Inputs	Expected Outputs
`11`	OR	`12` (`1100`), `10` (`1010`)	`14` (`1110`), `0`, `0`