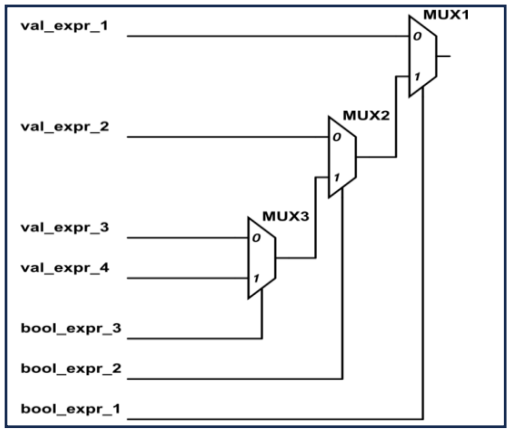
Difference Between case and if-else in Demux Implementation

Both case and if-else statements can be used to implement a demultiplexer in Verilog, but they differ in how synthesis tools interpret them.

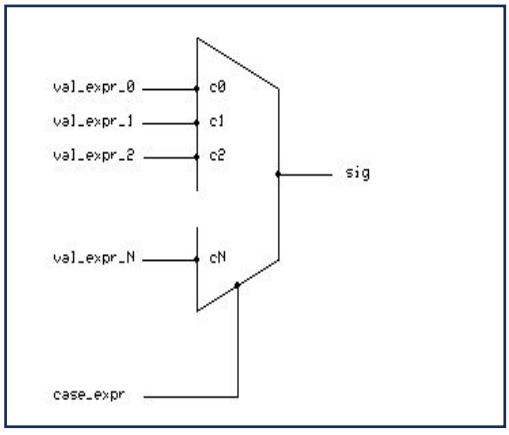
- The case statement is generally preferred for decoding logic like demuxes, as it does not assign priority to any condition. It leads to cleaner and more optimized hardware in most synthesis tools.
- The if-else statement introduces priority — the first condition has the highest priority. While functionally correct, this can lead to extra logic during synthesis and might not be as efficient.

In summary, use case when all conditions are mutually exclusive and equally important (like in a demux), and use if-else when priority or sequential decision-making is needed.

Aspect	`case` Statement	`if-else` Statement
Priority	No priority — all cases treated equally	Has implicit priority — top `if` has higher
Synthesis Result	Maps efficiently to decoder logic	May introduce priority logic
Hardware Logic	Cleaner multiplexer-style output	Potentially more complex logic gates
Best Use Case	Ideal for demux and address decoding	Better for conditional decision-making
Risk of Latch	Low, if all cases are covered	Medium, if `else` or default handling is missing



Using if-else



Using case statement