VHDL

Lab2

- **ALU Verification Using Assert Statements**.
  - **Objective:**
    - Design a comprehensive testbench for a given Arithmetic Logic Unit (ALU) module using VHDL ASSERT statements to validate its functionality.
  - **Requirements:**
    - Develop test cases that verify all ALU operations (e.x., ADD, SUB, Mul, Div) with varied input combinations.
    - Use ASSERT statements to:
    - Check output correctness against expected results.
    - Report failures with descriptive error messages (e.x., "ADD operation failed: Expected 0x5, Got 0x3").
    - Ensure the testbench covers edge cases (e.x., overflow, zero inputs).

- **FSM Verification Using File I/O**
  - **Objective:**
    - Implement a testbench for a Mealy-type Finite State Machine (FSM) with a 2-process architecture (mealy_2p), using file-based test vectors for input stimuli and result logging.
  - **Requirements:**
    - **Testbench Workflow:**
      - Read input stimuli (reset, input signals) from a text file (fsm_test_vectors.txt).
      - Apply inputs to the FSM synchronously with the clock.
      - Log outputs (or state transitions) to a results file (fsm_results.txt).
    - **File Formats:**
      - Input File: Space-separated values (e.x., reset input per line).
      - Output File: Record FSM outputs/state changes per clock cycle.
    - **Validation:**
      - Manually verify logged results against expected behavior.