Section 3 – FSM Design

Objectives

- Design and verify following FSM-based systems:
    1. **Sequence Detector FSM** (Moore or Mealy)
    2. **VGA Controller**
    3. **UART communication protocol**
    4. **I²C communication protocol**
- Develop functional coverage using **SystemVerilog covergroups**.

Tasks

---

1. Sequence Detector FSM

Design Requirements

Implement a sequence detector FSM (`seq_detector.sv`) that detects a specific bit pattern such as:

- `"1011"`

Specifications:

- Input: `clk`, `rst_n`, `in_bit`
- Output: `seq_detected`
- FSM style: **Moore or Mealy** (state diagram required)
- Must support:
    - **Overlapping sequence detection**
    - Clean reset behavior

Testbench Requirements

Create `seq_detector_tb.sv` that includes:

- Directed stimuli (patterns that hit overlapping and non-overlapping cases)
- Random bit-stream generation
- Expected output computation inside the testbench
- **Functional coverage**:
    - **State coverage** (every state hit)
    - **Transition coverage** (every transition taken)
    - **Sequence coverage** (cover consecutive input patterns)

---

2. VGA Controller FSM

Design Requirements

Design VGA Controller FSM using SRAM

---

3. UART Communication protocol FSM

Design Requirements

- Design UART Transmitter and receiver (already covered in lectures).
- Create reusable System Verilog interface for TX/RX signals.

---

4. I²C Communication protocol FSM

Design Requirements

Create a design module to transmit and receive 12-bit data via I2C protocol

Specifications:

- Data must transmit following strict I2C communication protocols. For example: after transmission of address and its acknowledgement from the slave, a data byte (8 bits) must transmit and its acknowledgement, followed by next byte (4bits + 0000) with zero padding.

---

5. Testbench

Write layered testbenches for all the above designs, provide functional coverage design.