

# Project Title: **Memory Game**

The system generates a random sequence of LED flashes that the player must replicate by pressing corresponding buttons. Each round, the sequence grows longer. If the player makes a mistake, the game ends and their final score, which is based on the round, will be displayed.

## **Technical**

### Core Functionality

- Pattern Generation: randomly generate a sequence of LED flashes, with each round adding a new step to the sequence
- Player Input: user will interact with the game by pressing buttons corresponding to LED positions
- Game logic: system will compare player's input against generated sequence to determine if input is correct
- Scoring: score will be tracked by the round number
- Display: the generated sequence will be shown on LEDs, and the current round number will be displayed on a seven-segment display

### Rough Design Descriptions

- FSM: handle the main game flow, managing states for pattern generation, user input collection, and error checking
- Pattern Memory: The sequence will be stored using a shift register
- input Handling: Buttons on the FPGA will be used for input
- Scoring System: The current round number (score) will be incremented with each successful round and displayed on a seven-segment display

## **I/Os**

### Inputs

- buttons for user input
- Reset button
- start button

### Outputs

- 8 LEDs for displaying the pattern
- 1 seven-segment display for showing the score/start

## Hardware Peripherals

- 7-segment display: To display the score/start
- 4 additional buttons for the user to input their guesses

## Module Header

```
module memory_game (  
    input logic clk, rst_n,  
    input logic [7:0] player_input,  
    input logic start,  
    output logic [7:0] leds,  
    output logic [7:0] score_display,  
);
```