

mayghhaa

Create x.md

4839dbf · 8 months ago

<div></div> Functional Table	Add files via upload	10 months ago
<div></div> Logisim Videos	Add files via upload	10 months ago
<div></div> Logisim	Add files via upload	10 months ago
<div></div> Snapshots	Add files via upload	10 months ago
<div></div> Verilog	Add files via upload	10 months ago
<div></div> Videos	Add files via upload	10 months ago
<div></div> README.md	Update README.md	10 months ago
<div></div> x.md	Create x.md	8 months ago

Memory Game

Team Details:

▼ Details

Semester: 3rd Sem B. Tech. CSE

Section: S1

1. Megha Arya 221CS136 meghaarya.221cs136@nitk.edu.in 7338430370

2. Sharada A Pandit 221CS151 sharadaapandit.221cs151@nitk.edu.in 9916411636

3. Viren Haresh Kundnani 221CS165 virenhakeshkundnani.221cs165@nitk.edu.in 8850778784

Abstract:

▼ Details

In our project, we'll be designing a sequential logic game, such as a memory game.Users must follow a sequence of LED flashes or button presses, and logic gates can control the game's logic. The initial sequence will be displayed with the help of LEDs. The player's input will be taken using pushbuttons. If the

input and the sequence matches, the next sequence flashes and the player is asked to either input the current or previous sequence to add to the challenge. In this project, we present the development of an interactive game system that creates a dynamic and engaging user experience. The primary objective of this project is to design and implement a digital circuit that displays a randomized sequence of LED patterns and validates user input to assess correctness. It offers a unique blend of hardware design, gaming, and user interaction, making it a compelling and engaging project, allowing for an exciting and variable gameplay. Key components of the project include modules for generating random sequences, controlling LEDs to display patterns, capturing user input, and implementing validation logic. We were motivated to do this project as in today's world full of stress, gaming is an outlet for people to relax. We are implementing a simple form of this relaxation for people to play while also testing their memory.

Working:

▼ Details

The project uses sequential circuits, i.e., D Flip Flops and a comparator circuit for the final output of whether the user was right or wrong. This output is also used as an enabler for the generation of the next sequence, hence if it is 0 the game does not continue. The random sequence is generated by using a Linear Feedback shift register, which generates all binary numbers with decimal values from 1 to 15 in random order. This is achieved by XOR'ing the last 2 bits of the previous sequence, shifting the 4 bits to the right (discarding the 4th) and placing this XOR value as the first bit for the new sequence. The user inputs the bits using buttons and the comparator output triggers the clock circuit for the next sequence to be generated. The LED's (2-bit : Either 01 or 10) are displayed using a D Flip Flop using the D and Q ends of the flip flop and the sequence sent to the comparator is decided using four 2:1 Multiplexers, each taking one bit of the current and previous sequences as the 2 inputs and LED output as the enabler. Based on the LED displayed to the user, one set of flip flops is used for generation and another set is used for storing the previous flashed sequence and depending on which one is asked, it is sent to the comparator circuit. The comparator simply consists of 4 XOR gates to output whether the input by the user and the sequence were the same.

Sr No.	Components	Functions
1	JK Flip Flops	Used to store both the previous and new sequence
2	Counter	Used to display the score in each level as well as the level the player is currently at.
3	LEDs	Used to display the

📖 README

✎ ☰

		whether to input the current sequence or the previous sequence Used to show whether the player has won or not
4	Multiplexer	Decides which sequence to input
5	XNOR gate	Used to compare the sequence inputted by the player and the correct sequence. If both are equal, 1 is given as the output
7	XOR gate	The last 2 bits in the sequence are XOR'ed to generate a new first bit so as to create a new sequence.
	Buttons	Used to take the input from the user.

Logisim Circuit Diagram:

▼ Details

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Contributors 5



Languages

● **Verilog** 100.0%