Lab 4: Mental Binary Math Game with ROM-based Game Access Control on FPGA

USER MANUAL

Mubashar Khan

ECE5440

# Introduction

The Mental Binary Math Game is a 1-player mental math game that tests the player’s understanding of binary and hexadecimal numbers. Players must be able to convert between binary and hexadecimal and add in binary to play this game.

The board has three main components that the players will be interacting with. There are four sections of four switches, four buttons and three 7-segment displays. Players must enter a predefined password in order to gain access to the game. Once the correct password has been entered, a green LED will turn on to let players know they can now access the game. The password can be entered using the switches and button marked with green. Player 1 will place inputs using the Player 1 switches and Player 1 button marked in red. Player 2 will place inputs using the Player 2 switches and Player 2 button marked in blue. Figure 1 shows these areas marked. The button marked in brown is the reset button. This will allow players to reset the game and log players out. From left to right, the first display will show Player 1’s input as a hexadecimal number. The second display will show Player 2’s input as a hexadecimal number. The third display will show the sum of Player 1’s and Player 2’s inputs as a hexadecimal number.

**A circuit board

Description automatically generated**

**Figure 1.** DE2-115 FBGA Board.

# Instructions for game operation

This is a turn-based game. To begin, one player must enter the password using the green switches displayed in figure 1. The password is a 4-digit number. For this game, the password is 2949. Using the switches, players can enter 1 digit at a time as a binary 4-bit number. Once a player uses the switches to enter a 4-bit value, the player must press the green button to enter the next digit. Repeat this process for all 4 digits and the Red LED will turn Green once players enter the password correctly.

Player 1 will pick a number between 0000 and 1111 in binary and input that into the board using the switches designated for Player 1. The board will convert the binary input into a hexadecimal number that will appear on Player 1’s 7-segment display. Player 2 should look at this hexadecimal number and then using the Player 2 switches, input the binary number that they think will add with Player 1’s number to get to 1111 in binary (F in hexadecimal). Difficulty can be increased by implementing a time limit on Player 2’s turn. We recommend 5 seconds.

**Password Entry Example:**

The password is 2949. Each digit of this password must be entered individually. 2 is 0010 in binary. We enter that value using the purple switches and then we press the purple button. The next number 9 is 1001 in binary. Again, we enter this value using the purple switches and the press the purple button. The next value is 4 which is 0100 in binary. We enter the value using the purple switches and then press the green button. The last digit in our 4-number password is 9, which is 1001 in binary. We enter this number using the purple switch and purple button. The red LED will turn off and the green LED will turn on.

**Digit Timer Setup Example:**

Let’s say we want to play for 60 seconds. Once we have entered our password, we need to enter the time we want using the gold and yellow switches. Because the time we want is 60 and each decimal place is represented by one 7-segment display, we split 60 seconds into a 6 and a 0. For the gold switch we enter 0110 and for the yellow switch we enter 0000. Then we press the purple password button to lock our time in. Once you are ready to start, simply hit the purple password button one more time. This will start the count down and let you begin playing the game.

**Example 1: Failed Round**

The player hits the red random number button to display a random number between 0-F. Let’s say it displays a 7.

The player decides they want to input 1010 in binary. They type it in with the green switches, and the green display shows the letter A in hexadecimal.

The sum displayed is 1. 0111+1010 = 10001, however, because the 7-segment display only works for 4-bit numbers for this game, the most significant digit is cut off. This is not F (1111 in binary) and so the player loses the round and does not get a point this round.

**Example 2: Successful Round**

The player hits the red random number button to display a random number between 0-F. Let’s say it displays a 9.

The player decides to enter 0110 into the green player input switches. The display shows 6

The sum displayed is F. That is binary 1111 and so the player gets a point. The player can continue until they run out of time or hit reset.