

**PAF KARACHI INSTITUTE OF ECONOMICS AND TECHNOLOGY**

*Department of Software Engineering*

**PROJECT REPORT**

***Simon Memory Game***

**Course Title:** Embedded System Design Lab

**Course Code**: 113498

**Class**: BESE

**Due Date: 4 Jan 2024**

**Course Instructor:** Engr. Farhan Ghafoor

Engr. Rehan Abu Hashir

**Lab Instructor:**

|  |  |  |
| --- | --- | --- |
| **GROUP MEMBERS** | | |
| **S.No** | **SID** | **Name** |
| **1.** | **14394** | **M.HAMZA** |
| **2.** | **14230** | **ASAD RASHEED** |
| **3.** | **12998** | **HASEEB ALI** |

1. **INTRODUCTION:**

Simon Memory game is based on short term memory skill. It is an electronic game that works by flashing LEDs and creating sounds to first memorize a sequence then press the corresponding buttons. It repeats the sequence by adding difficulties like speed and order of flashing LEDs. If you memorize the sequence and press the correct buttons you proceed to the next level and if you fail to press the correct buttons, the game is over. Some Simon Memory games are also based on time limits in which for a particular time you must complete all the levels and if the time runs out, the game is over. This type of can improve skills like enhancing memory and improving reaction time.

Our implementation of this game includes two types of modes and a score keeping system to make this game more interesting to play.

1. **HARDWARE:**

**Embedded Board Description:**

The board we selected to implement our project on is Arduino UNO R3. It consists of AVR (Advanced Virtual RISC) architecture. It is a microcontroller with speed of 16MHz and has 6 analog inputs along with 14 digital pins from which 6 provides PWM output. We can’t program Arduino UNO on our normal windows environment, and we must use Arduino IDE to configure and program our system.

**Input Device Description:**

The game uses 4 input buttons with different colors like red, yellow, green and blue to get the corresponding input from the user. These buttons are connected to the digital pins of Arduino UNO from 2 to 5.

**Output Device Description:**

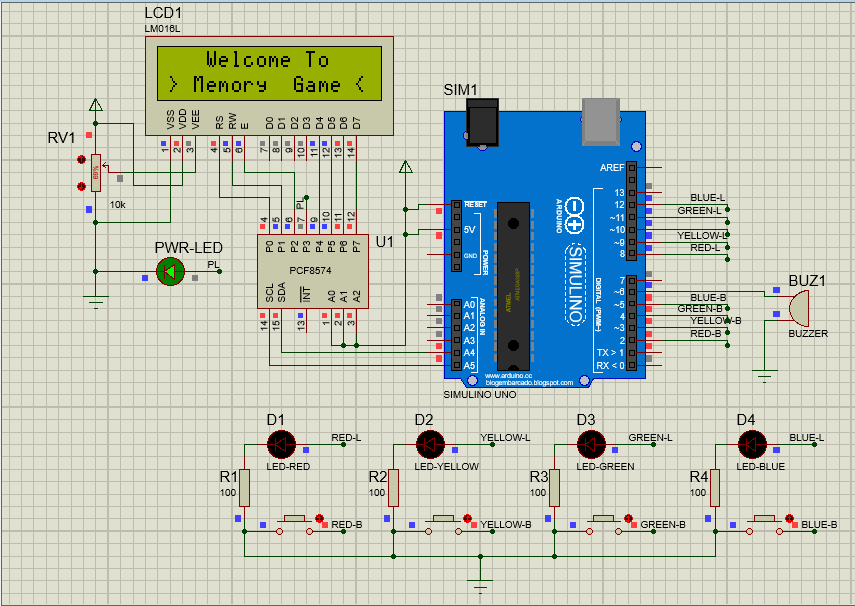
According to the inputs, the game uses 4 LEDs to show the output and uses a buzzer to create specific sounds for each LED. It also has a 16x2 LCD display to show the progression of player and other information. The LEDs are from 8 to 11 while the buzzer is at pin 6. The 16x2 LCD display have I2C module with it so it is connected to the analog pins, A4 and A5 of Arduino UNO.

**Schematic Diagram & Links:**

Arduino UNO R3 schematic Link: <https://www.arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf>

16x2 LCD with I2C module schematic Link:

<http://handsontec.com/dataspecs/module/I2C_1602_LCD.pdf>

Below is our schematic of our project which shows the above-mentioned connections:  
  


**Power Consumption:**

The Arduino UNO R3 consumes power of 5v to run but in our case, it takes 9v. We can also use the Arduino UNO USB cable to take power, which is more efficient as with bigger hardware it consumes more power so 9v battery would be dead in a few runs.

1. **SOFTWARE:**

**Programming Language:**

The Simon Memory Game is implemented using the Arduino programming platform, specifically in Arduino C++. The selection of C++ as the programming language was a strategic decision influenced by several factors:

Reasons for Choosing C++:

* C++ offers a higher-level abstraction compared to assembly language, making it more accessible for rapid development and prototyping.
* C++ allows for quick implementation of algorithms and logical structures, facilitating the development process.
* C++ strikes a balance between readability and efficiency, making the code more understandable for developers at various skill levels.
* C++ provides a rich set of features and libraries that enhance the functionality of the Simon Memory Game without the need for low-level assembly commands.

While assembly language could provide more granular control over the microcontroller, the trade-off for ease of development and enhanced functionality led to the selection of C++ for our project. This choice aligns with the project's goals and the diverse skill set of the development team.

**Real Time Constraints:**

The Simon Memory Game operates without stringent real-time constraints. The gameplay involves user input response within acceptable time frames, and delays are carefully calibrated to ensure a smooth and engaging user experience. Unexpected delays may result in a minor disruption to gameplay but do not compromise the overall functionality.

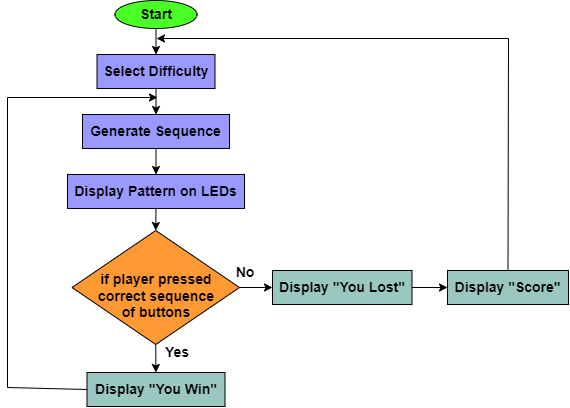
**Security:**

In the context of the Simon Memory Game, there are minimal security concerns. The nature of the application, being a game, doesn't involve sensitive data or interactions that could be exploited. As such, the security posture is inherently low risk.

**Code Attribution:**

The code includes an header file of “**LiquidCrystal\_I2C.h**” which is an online library for 16x2 LCD with I2C module to provide easy access to LCD display commands and this can also be found online. The code is at most original to the group as we have added additional features to a normal Simon Memory Game like a score system which keeps the players score and also records the high score and a game mode system which includes two modes of normal and hard with different functionalities to make the game more challenging.

1. **FLOWCHART:**



1. **RELATED WORK:**

While the Simon Memory Game is a classic and widely recognized concept, the specific implementation details may vary across different projects. However, our project distinguishes itself through the incorporation of additional features, such as a scoring system, high score tracking, and multiple difficulty modes. These enhancements contribute to a more engaging and challenging gameplay experience for the users.

1. **CONCLUSION:**

The Simon Memory Game project provided valuable insights into embedded systems, game development, and collaborative problem-solving. The challenges faced and overcome during the development process enriched our learning experience and underscored the importance of effective teamwork. Here are some major conclusions of our work:

**Group Contribution:**

Our group has contributed equally in every part of making this game. This shows how much effort we have equally put into creating this game and with each other’s equal help we have made it faster and more efficient.

**Challenges:**

The challenges we have faced are making this game with a functionality of difficulty curve. A difficulty curve means that we must make a system which changes with each iteration, increasing difficulty with each level. This also means the system must check the modes too that what has been chosen by the player and then set the difficulty curve accordingly.

**Future Work:**

More improvement in the future could be made to the user interface by using different kinds of LCDs like providing a touch display or having animations, a menu for different operations and more visuals in the display. Another thing could be to link it with Bluetooth or Wi-Fi modules to make a multiplayer real-time game where a group of friends can compete.