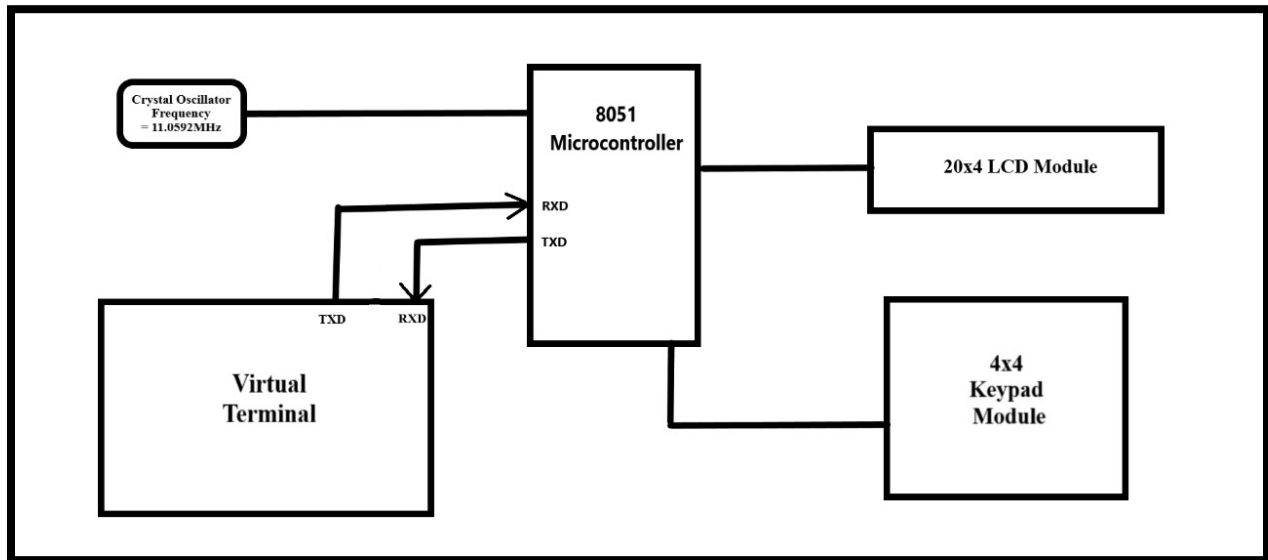


AIM:

The aim of this project is to create a quiz game using 8051 which has two modes, single player and two players.

BLOCK DIAGRAM:



The components used are:

- 8051 Microcontroller (AT89C51)
- 20×4 LCD Module
- Keypad
- Serial Monitor
- Resistors
- Capacitors
- Reset button

DESCRIPTION ABOUT THE COMPONENTS:

1. 8051 Microcontroller:

In 1981 Intel designed 8051 Microcontroller. It is an 8-bit microcontroller. It is built with 40 pins DIP (dual inline package), 4kb of ROM storage and 128 bytes of RAM storage, 2 16-bit timers. It consists of are four parallel 8-bit ports, which are programmable as well as addressable as per the requirement. An on-chip crystal oscillator is integrated in the microcontroller having crystal frequency of 12 MHz.

➤ *Architecture of 8051 Microcontroller:*

- The 8051 Microcontroller is an 8 – bit Microcontroller i.e. it can read, write and process 8 – bit Data. There are a bunch of manufacturers like Atmel, NXP, TI, who manufacture their own versions of 8051 Microcontroller.

- Irrespective of the manufacturer, the internal hardware design i.e. the 8051 Microcontroller Architecture remains more or less the same. The block diagram of the 8051 Microcontroller Architecture shows that 8051 Microcontroller consists of a CPU, RAM (SFRs and Data Memory), Flash (EEPROM), I/O Ports and control logic for communication between the peripherals.
- All these different peripherals inside the 8051 Microcontroller will communicate with each other via the 8 – bit Data Bus, also known as the internal data bus.

➤ **Fundamental components present internally inside 8051 Microcontroller architecture are:**

a. **CPU (Central Processing Unit):**

CPU act as a mind of any processing machine. It synchronizes and manages all processes that are carried out in microcontroller. User has no power to control the functioning of CPU. It interprets the program stored in ROM and carries out from storage and then performs it projected duty. CPU manage the different types of registers available in 8051 microcontroller.

b. **Interrupts:**

Interrupts is a sub-routine call that given by the microcontroller when some other program with high priority is request for acquiring the system buses the n interrupts occur in current running program. Interrupts provide a method to postpone or delay the current process, performs a sub-routine task and then restart the standard program again.

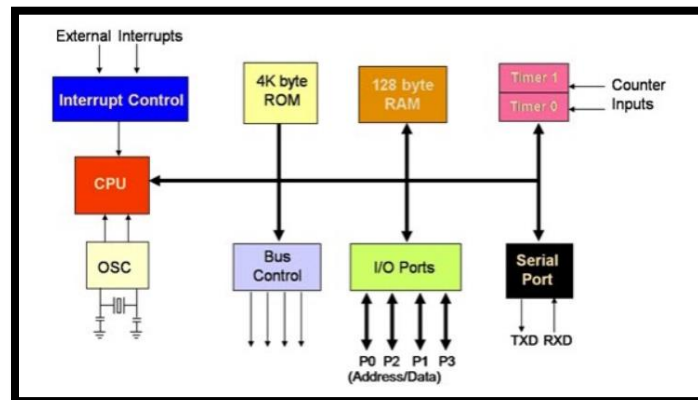


Fig 1: Architecture of 8051 Microcontroller

c. **Memory:**

For operation Micro-controller required a program. This program guides the microcontroller to perform the specific tasks. This program installed in microcontroller required some on chip memory for the storage of the program. Microcontroller also required memory for storage of data and operands for the short duration. In microcontroller 8051 there is code or program memory of 4 KB that is it has 4 KB ROM and it also comprise of data memory (RAM) of 128 bytes.

d. **Bus:**

Bus is a group of wires which uses as a communication canal or acts as means of data transfer. The different bus configuration includes 8, 16 or more cables. Therefore, a bus can bear 8 bits, 16 bits all together.

e. **Oscillator:**

As the microcontroller is digital circuit therefore it needs timer for their operation. To perform timer operation inside microcontroller it required externally connected or on-chip oscillator.

Microcontroller is used inside an embedded system for managing the function of devices. Therefore, 8051 uses the two 16 bit counters and timers. For the operation of this timers and counters the oscillator is used inside microcontroller.

2. LCD Module (20 x 4):

The LCD stands for liquid crystal display, that works on the light modulation features of liquid crystals. It is available in electronic visible display, video display and flat panel display. There are numerous categories and features are exits in markets of LCD and you can see it on your mobile, laptop, computer and television screen.

➤ *Features of the 20 x 4 LCD Display:*

- In a 20x4 LCD module, there are four rows in display and in one row twenty character can be displayed and in one display eighty characters can be shown.
- This liquid crystal module uses HDD44780 (It is a controller used to display monochrome text displays) parallel interfacing.
- The liquid crystal display interfacing code is easily accessible. We just required eleven input and output pinouts for the interfacing of the LCD screen.
- The input supply for this module is three volts or five volts, with that module other components like PIC, Raspberry PI, Arduino.
- This electronic device can be used in different embedded systems, industries, medical devices, and portable devices like mobile, watches, laptops.
- Liquid crystal display works on two types of the signal first one is data and the second one is for control.
- The existence of these signals can be identified through the on and off condition of RS pinout. Data can be read by pushing the Read/write pinout.

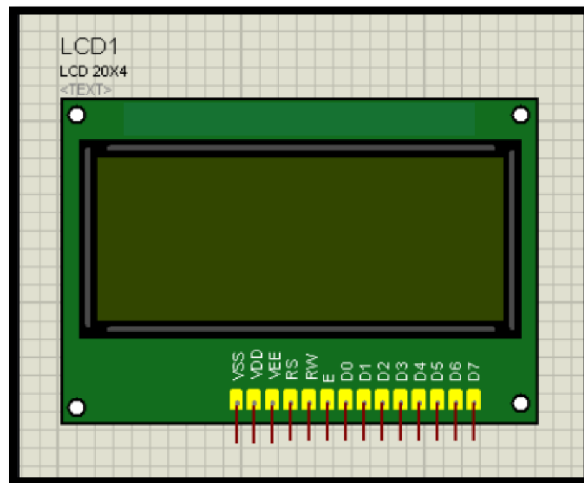


Fig 2: 20x4 LCD Display

3. Keypad Module

Keyboards are organized in a matrix of rows and columns

- The CPU accesses both rows and columns through ports
- Therefore, with two 8-bit ports, a 4 x 4 matrix of keys can be connected to a microprocessor
- When a key is pressed, a row and a column make a contact
- Otherwise, there is no connection between rows and columns

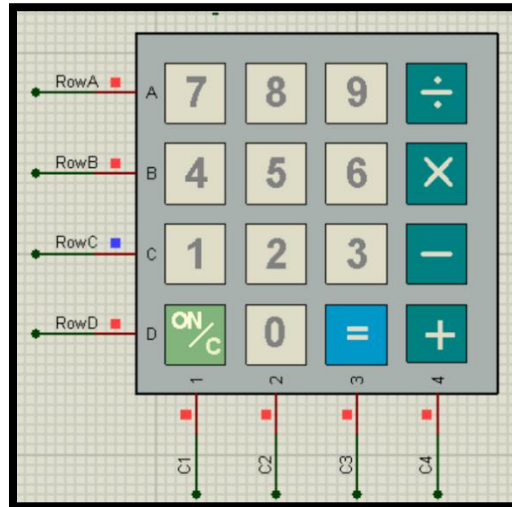
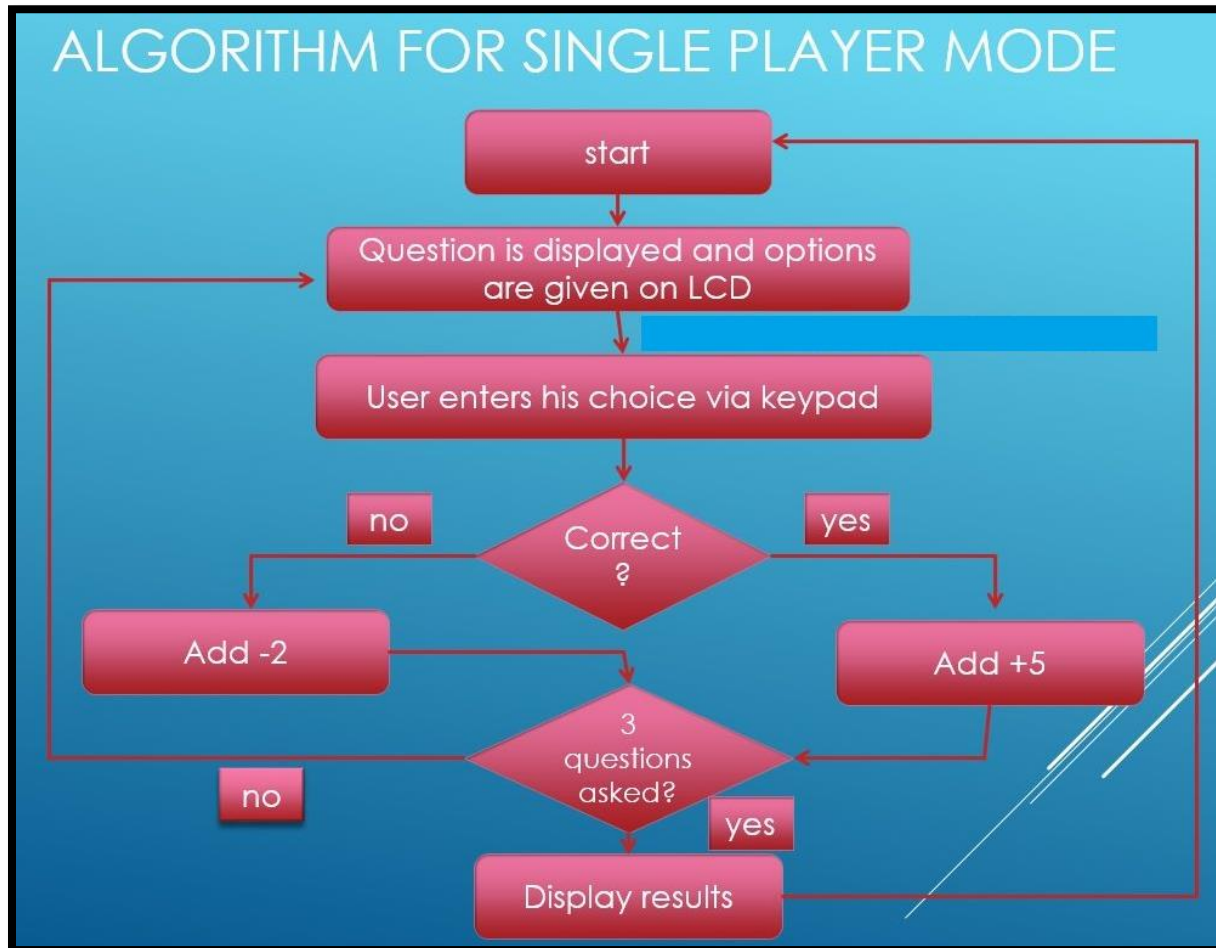


Fig 3: Keypad Module

ALGORITHM:

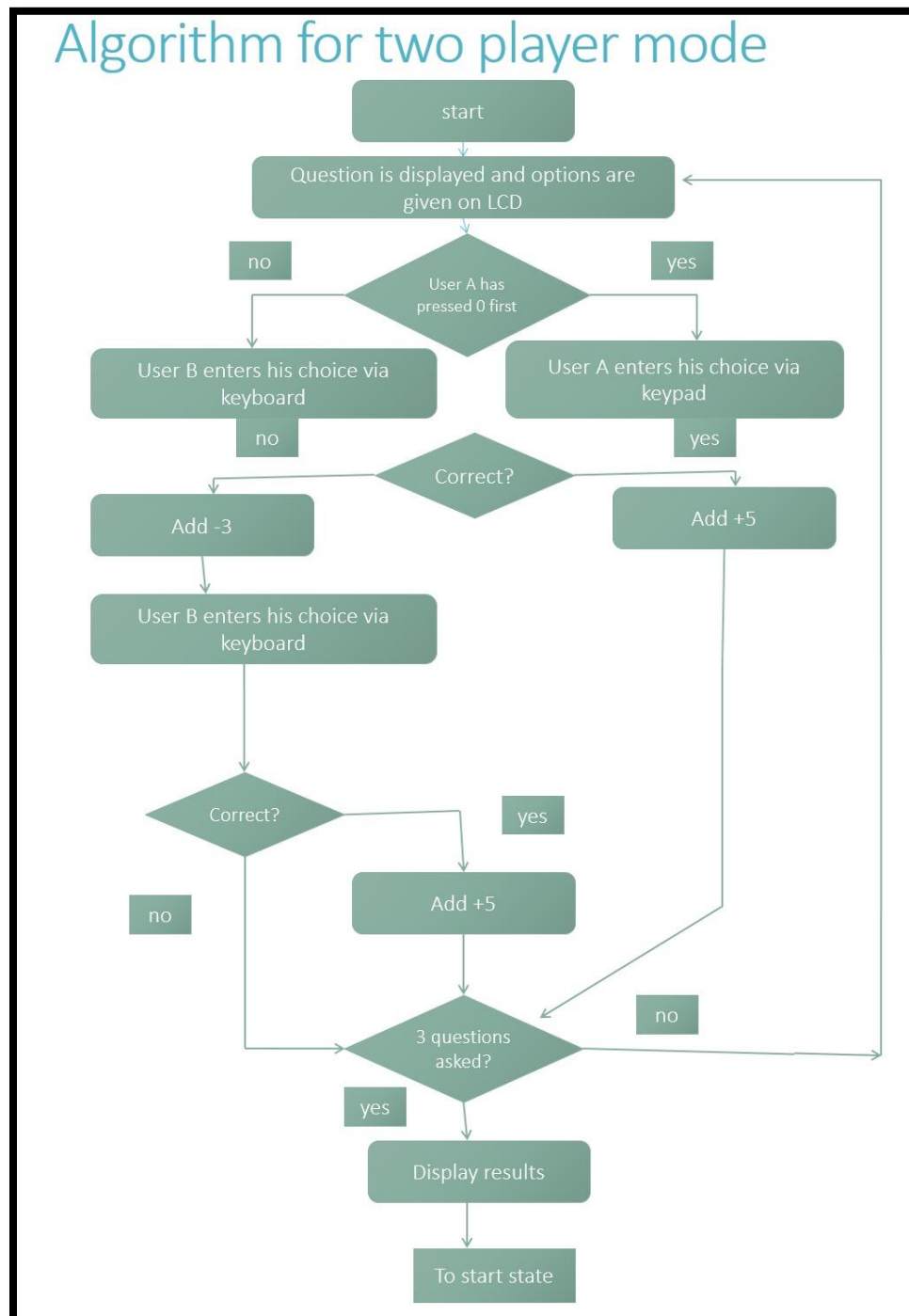
a. Single Player Mode -



- ❖ When user selects mode 1 i.e. single player mode, then the questions will be displayed on the LCD and the questions can be answered using the keypad that is connected.
- ❖ The LCD waits until the user gives an answer.
- ❖ If the user gives the correct answer then "Correct" will be displayed otherwise "Wrong" will be displayed.
- ❖ After that the LCD waits for 3 seconds and then displays the next question. This process is continued until all 5 questions are displayed.
- ❖ After all the questions are finished then the LCD waits for 5 seconds and then returns to the initial page where the user can select the mode once again and continue playing if he wants.
- ❖ If you press the reset button then the system will be restarted.
- ❖ Various functions are used in this mode:
 - cct_init() – This function initializes all the port pins when the microcontroller is turned on.
 - lcd_init() – This function is used to initialize the LCD.
 - writemcmd() – This function is used for clearing LCD screen, moving the cursor, etc.

- `writeline_lcd()` – This function is used for displaying text in the LCD display.
- `get_key()` – This function is used for receiving the value of the button that is pressed on the keypad.
- `delay()` and `high_delay()` – These are the functions that are used for time delay operations in the program.

b. Two Player Mode -



- ❖ If the user selects mode 2 i.e. two player mode, then the questions will be displayed on the LCD as well as Virtual Terminal both.
- ❖ The player who is playing the quiz game using the LCD can give answers through the keypad.
- ❖ The player who is playing the quiz game using the Virtual Terminal can give answers through the keyboard on his PC/Laptop.
- ❖ If the user gives the correct answer then “Correct” will be displayed otherwise “Wrong” will be displayed.
- ❖ After that the LCD and Virtual Terminal displays the next question. This process is continued until all 5 questions are displayed.
- ❖ After all the questions are finished then the LCD and Virtual Terminal returns to the initial page where the user can select the mode once again and continue playing if he wants.
- ❖ If you press the reset button then the system will be restarted.
- ❖ Various functions are used in this mode:
 - SerialInitialize() – This function is used to initialize the serial communication pin in the 8051 microcontroller by setting the timer up, setting up SCON register and setting the baud rate.
 - uart_tx() – This function is used to give commands to the virtual terminal. Using this we can move the cursor to the next line in the virtual terminal by passing a value of 0x0d to the function.
 - writeline_serial() – This function is used for displaying text in the virtual terminal.
 - receive() – This function is used to receive the values that we entered into the virtual terminal.

PROTEUS SIMULATION:

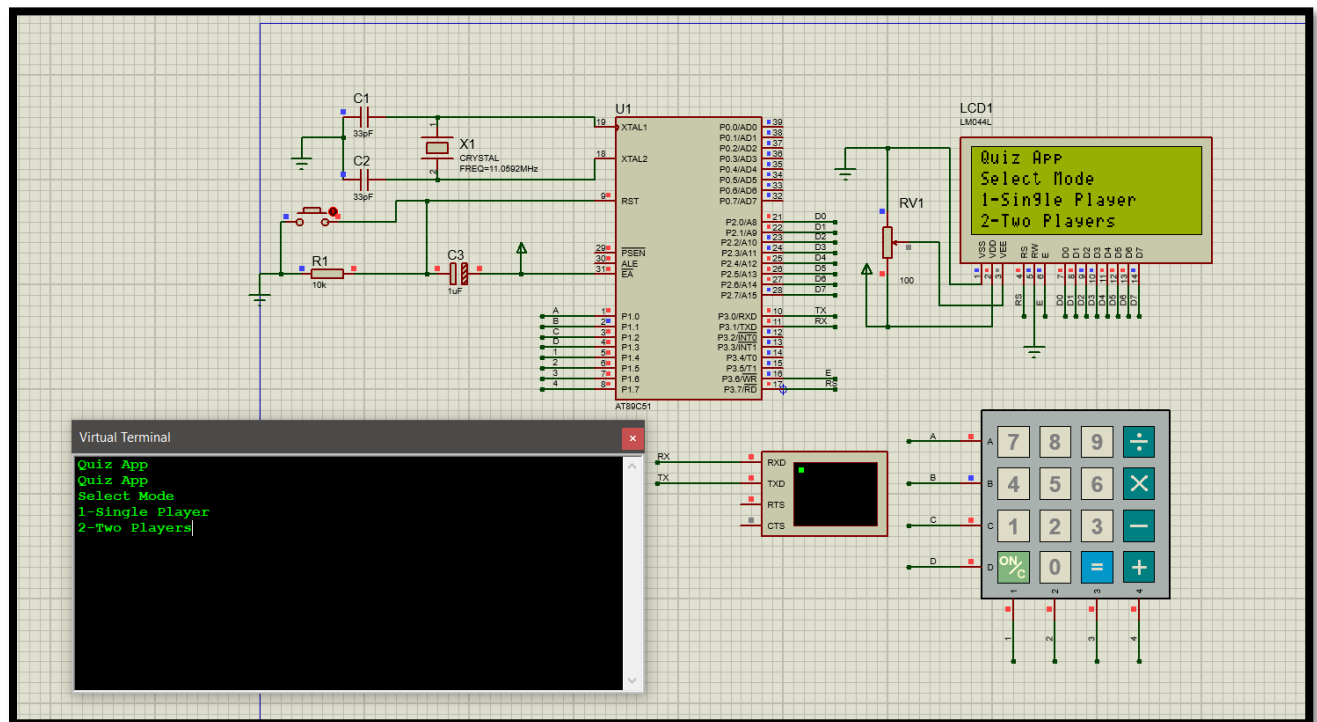


Fig 4: Initial page of the game

PROS AND CONS:

➤ Pros:

- ❖ Low time required for performing operation.
- ❖ The processor chips are very small and flexibility occurs.
- ❖ Due to their higher integration, cost and size of the system is reduced.
- ❖ The microcontroller is easily to interface additional RAM, ROM and I/O ports.
- ❖ Once microcontrollers are programmed then they cannot be reprogrammed.
- ❖ At the same time many tasks can be performed so human effect can saved.
- ❖ Without any digitals parts it can be act as microcomputer.
- ❖ It is easy to use, troubleshooting and systems maintain is simple.

➤ Cons:

- ❖ Maximum only two players can play this quiz game because we connect a maximum of 3 devices to the 8051 Microcontroller. Even though in our project one port is free we cannot use it because we need at least two ports to be free if we want to add another LCD and Keyboard.
- ❖ In our project we are using both Virtual Terminal and LCD both but the questions that are being displayed are not being updated properly after each question.
- ❖ When we select 2 player mode, questions which are displayed do not get updated until we give an answer in the LCD.
- ❖ The Virtual Terminal does not let LCD player give the answer first. Only after the player 2 gives an answer then player 1 can give his answer.
- ❖ This can be avoided by tweaking the code a little bit.
- ❖ Another problem is that if the code is too large (size is greater than 540 lines) then the HEX file will not be generated by Keil and hence we cannot operate this quiz game.
- ❖ As a result, we were not able to display the marks of the player after their game.

FUTURE WORK:

- A buzzer system can be added to this project so that we can use this in quiz competitions with a very high accuracy and low margin of error. Whoever presses the buzzer first will get the chance of answering and if the answer is wrong the question is passed to the next player.
- If possible, we can link this project to a database that can randomly pick questions for each player and thus ensuring that there is a low chance of malpractice. In this case the microcontroller randomly picks 5 questions for each player by itself and displays it on the display so that the players can answer. Checking of the answers will also done in a similar way where the microcontroller compares the answers given by the players with the answer that is fed into the database.
- We can also include a marking scheme so that instead of showing if the answer is right or wrong after each question, the correct answers will be counted and thus marks will be given to the players.
- If we want, we can replace the LCD module with a better display in order to make the interface look even more engaging.