CSE231: Microprocessor Systems and Interfacing Project — PONG GAME

Spring 2024

1*st* Student name: Doha Bahaaeldin Hafez
ID: 21-101136

2*nd* Student name: Youssef Yasser
ID: 21-101115

3*rd* Student name: Aya Wael
ID: 21-101179

4*th* Student name: Jana Gamal
ID: 21-101104

5*th* Student name: Farah Tamer
ID: 21-101050

1. Introduction

This report details the development of a Pong game created using an STM32 microcontroller, LEDs, and two buttons for multi-player input. This project aimed to apply microcontroller programming skills and design principles to create an interactive and fun application.

2. Project Overview

The Pong game is a simplified version of the classic arcade game. It involves two paddles controlled by players, which are used to hit a moving ball back and forth. In this project:

- The ball is represented by LEDs.
- Two buttons are used for multi-player input to control the paddles.
- Buzzer to simulate scoring.
- The STM32 microcontroller handles the game logic, including ball movement and collision detection.

3. Hardware Components

Microcontroller

- STM32F103C8T6: The microcontroller used for this project. It is part of the STM32 family and is known for its versatility and performance in embedded applications.

LEDs

- LED Array: An array of 7 LEDs arranged to represent the ball. Each LED can be individually controlled to simulate the movement of the ball.

Buttons

- Push Buttons: Two push buttons were used as input devices for the players to control the movement of their respective paddles.

Buzzer

- Two buzzer: buzzers are used to beep when someone out of the two players scores.

Other Components

- Resistors: Used to pull the buttons down when they are not pressed to ensure no garbage value gets to the buttons.
- Breadboard and Jumper Wires: For building the circuit and connecting the components.

4. Game Logic

The software was designed to handle the following game elements:

- The game loop involves toggling the LEDs to represent ball and paddle movement.
- The ball starts at player one's side and moves back and forth, toggling different GPIO pins to represent its position.
- Button presses are read to check for paddle interactions.
- Buzzer to announce the scoring of any of the players.
- When someone scores 2 points, the speed of the game (ball movement) increases.
- When one of two players achieves a score of 5 points, the game terminates after beeping and powering the two red LEDs.

5. Software Design

Development Environment

- IDE: STM32CubeIDE and STM32CubeProgrammer
- Programming Language: C

Code Structure

The main code is structured to perform the following steps in a continuous loop:

Initialize System and Peripherals:

  - HAL_Init(): Initializes the HAL Library.
  - SystemClock_Config(): Configures the system clock.
  - MX_GPIO_Init(): Initializes all configured GPIO peripherals.

Ball and Paddle Movement

The ball's position is updated based on its current direction and speed:

- Ball Movement: Implemented using HAL_GPIO_WritePin() to set and reset GPIO pins corresponding to the ball's position.
- Paddle Control: Paddle movements are checked using HAL_GPIO_ReadPin() to detect button presses, and LEDs are toggled accordingly.

Game Control

- Buttons: Two buttons connected to GPIO pins BUTTON_B8_PIN and BUTTON_B9_PIN are used to control the paddles.
- Collision Detection: When a button is pressed, it triggers a change in the ball's direction if the ball is in the correct position.

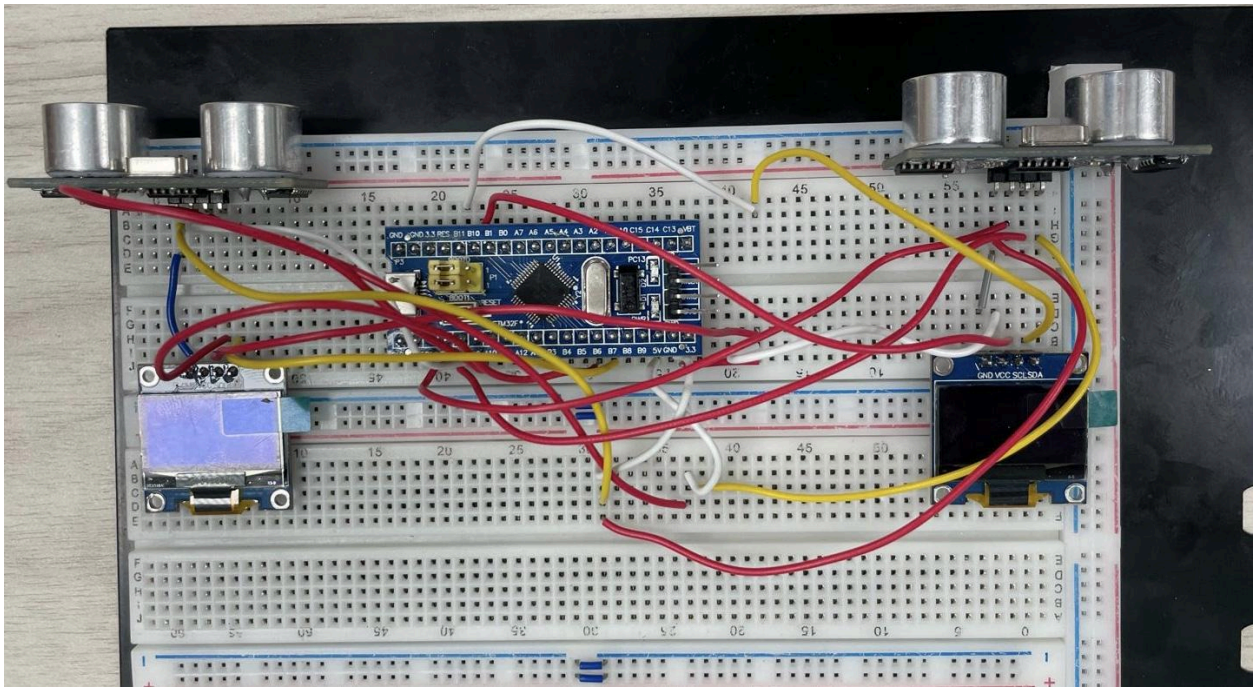
6. Challenges and Future Improvements

- **Game Display on LCD**

Displaying the game on an LCD would make the end-result more interesting and would be a great future improvement. We tried to turn on the LCD present in the ARM Development Panda Kit and we were able to; However, we could only print text.

- **Utilizing Ultrasonic Sensor**

Using the ultrasonic sensor to simulate paddles with our hand movement was a major step in our project and we worked hard to get to the point where we can measure distances using the sensor; However, we weren't able to connect it with the LEDs, yet it would make a great future improvement as well.



- **Seven-Segment Display for Scoring**

To show the scores of each player, we were originally using the seven-segment display in the ARM Development Panda Kit for that. Despite writing the code successfully, there was still no response from the kit. Using the seven-segment display in the future as a way to show the scores of the players would make the game more appealing.

7. Conclusion

The Pong game project provided valuable experience in microcontroller programming. It successfully demonstrated the use of an STM32 microcontroller to create an interactive game using LEDs and buttons. Despite the challenges, the final implementation was successful and users found the game to be engaging and fun.

8. References

- https://youtu.be/BK-bL9cMOSM?si=Sabtl-0FOKdg1yCO
- https://embetronicx.com/tutorials/microcontrollers/stm32/stm32-gpio-tutorial/
- https://youtu.be/YVbZR3-EoKM?si=u-cvUO9bwxrzZPCI
- https://youtu.be/e_NSqz5P8Qk?si=IzErJ9FiFiTKShz5
- https://youtu.be/oIbFqktTuuE?si=U18qhEVZQ-F5du1x