KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY
COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING

MICROPROCESSORS

GROUP 17

NANA AKOSUA ADDIPA – 1813522

TEMILOLUWA IFEDAPO TITILOYE – 1828822

GABRIEL NII ATTOH QUAYE – 1827522

ASARE KINGSLEY DONKOR – 1818122

BISMARK AMOAH – 1816222

RANDY AGYEKUM – 1814622

DADZIE KWESI ODARTEY - 1820722

SABLAH FREDRICK – 7108821

OWUSU TAKYI DANIEL – 7107621

MARFO RICHMOND MENSAH - 7102721

ABSTRACT

Microprocessors and microcontrollers are essential components in modern computing and embedded systems. A microprocessor is the central unit of a computer system, designed primarily for processing tasks and executing instructions, making it suitable for applications requiring high computational power, such as personal computers and servers. In contrast, a microcontroller integrates a processor, memory, and peripherals into a single chip, making it ideal for embedded systems, including consumer electronics, automotive applications, and industrial automation. While microprocessors excel in complex computing tasks, microcontrollers are optimized for real-time control and low-power applications.

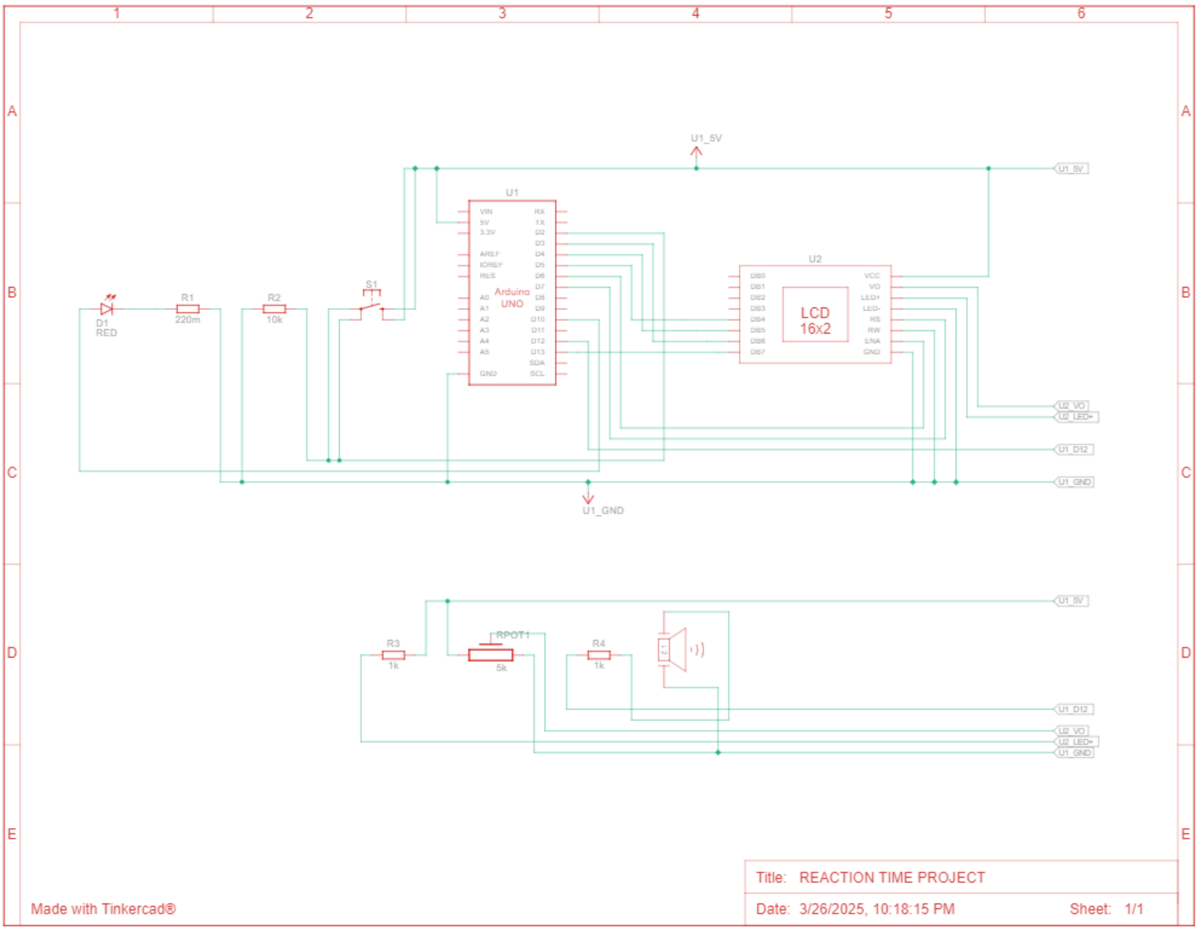
TABLE OF CONTENTS

INTRODUCTION

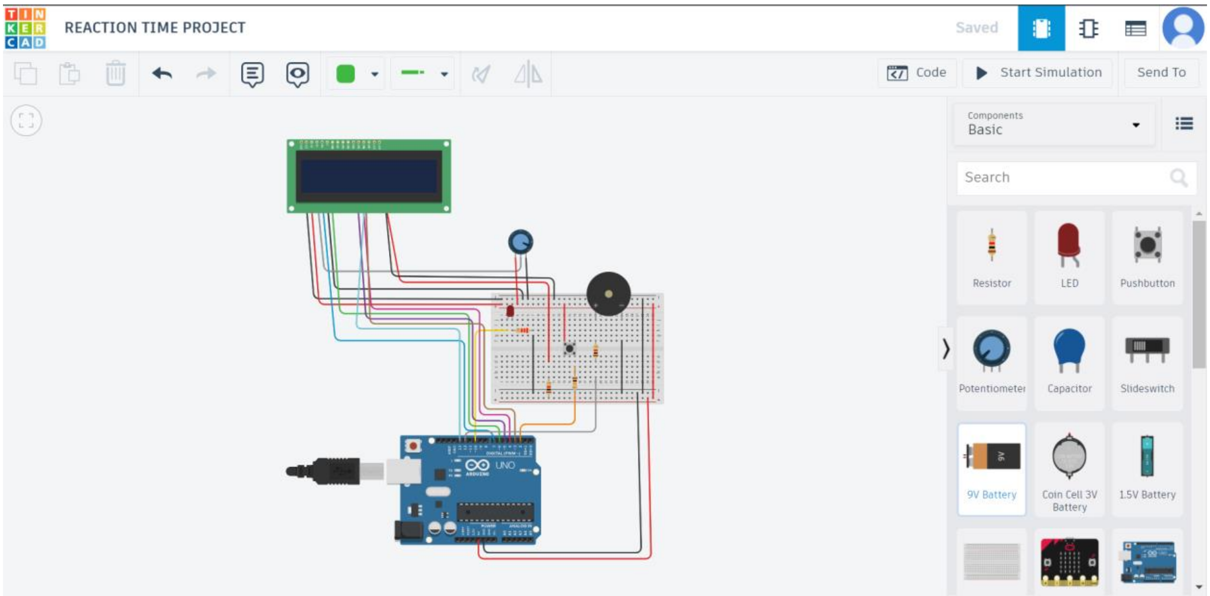
In this project, we aim to design a “Reaction Time Tester”.

Reaction time is a significant measurement of human reflexes and mind processing speed. The “Reaction Time Tester” is an interactive game whose goal is to measure and improve the response time of a player. The game operates by randomly illuminating an LED, which encourages the player to press a button as quickly as possible after recognizing the light. The apparatus then measures the reaction time and shows the reading on an LCD screen. The simple yet effective design provides an intriguing way of measuring reflexes as well as having potential applications in fields such as neuroscience, sports training, and human-computer interaction studies.

SCHEMATIC



CIRCUIT CONFIGURATION



COMPONENTS

Name	Quantity	Component
U1	1	Arduino Uno R3
S1	1	Pushbutton
R2	1	10 kΩ Resistor
R1	1	220 mΩ Resistor
D1	1	Red LED
U2	1	LCD 16 x 2
Rpot1	1	5 kΩ Potentiometer
R3 R4	2	1 kΩ Resistor
PIEZO2	1	Piezo

METHODOLOGY

Pin Configuration

LCD (16x2):

- RS: pin 7
- E: pin 6
- D4: pin 5
- D5: pin 4
- D6: pin 3
- D7: pin 13

LED: Connected to pin 10

Push Button: Connected to pin 2 (supports interrupts)

Buzzer: Connected to pin 12 (for sound feedback)

Game States

The game operates in five distinct states:

1. Idle Mode (0): Waiting for the player to press the button to start.
2. Waiting Mode (1): Random delay period before the LED turns on.
3. React Mode (2): LED is on, waiting for the player to press the button.
4. Game Ended Mode (3): Displays the player's reaction time and best score.
5. Timeout/Cheat Mode (4): Activated when the player presses too early or fails to react in time.

How to Play

1. Press the button to start the game (a "ready" tone will play).
2. Wait for the LED to turn on (random delay between 2-6 seconds).
3. When the LED lights up, a distinct tone will play.
4. Press the button as quickly as possible once the LED lights up.
5. Your reaction time will be displayed on the LCD.
6. If you achieve a new best time, a special high score melody will play.
7. Press the button again to play another round.

Features

- Interrupt-based Button Detection: Uses hardware interrupts for responsive button detection.

- No-Rush Mechanism: Detects if the button is pressed before the LED turns on.

- Timeout Function: Ends the game if the player doesn't react within 5 seconds.

- High score Tracking: Keeps track of the fastest reaction time.

- Debouncing: Prevents false button presses.

- Sound Feedback: Various musical tones and melodies provide audio cues for different game events:

  - Startup melody when the game initializes
  - Ready tone when a new game starts
  - LED-on indicator sound when it's time to react
  - Success melody when player successfully reacts
  - Special melody for new high scores
  - Failure tone for rushing or timing out

Sound Implementation

The game uses Arduino's `tone () ` function with a piezo buzzer for audio feedback:

Musical Notes:

- C5 (523 Hz), E5 (659 Hz), G5 (784 Hz), C6 (1047 Hz).

- A4 (440 Hz), B4 (494 Hz), F5 (698 Hz).

Sound Events:

- Game Startup: C-major arpeggio (C5-E5-G5-C6).

- Game Ready: G5 note.

- LED On: A4 followed by B4.

- Successful Reaction: Quick C5-E5-G5 arpeggio.

- New High Score: Extended C-major arpeggio.

- Failure (rush/timeout): Descending E5-C5 notes.

Customization Options

You can easily customize the game by modifying these constants in the code:

- `DISP_REFRESH_INTERVAL`: How often the LCD refreshes (in ms).
- `DEBOUNCE_TIME`: Minimum time between valid button presses (in ms).
- `TIMEOUT_DURATION`: How long the player has to react (in ms).
- `MIN_WAIT_TIME`: Minimum random delay before LED turns on (in ms).
- `MAX_WAIT_TIME`: Maximum random delay before LED turns on (in ms).

Sound Customization:

- `TONE_DURATION_VERY_SHORT`: Duration for very brief tones (100ms).
- `TONE_DURATION_SHORT`: Duration for short tones (150ms).
- `TONE_DURATION_MEDIUM`: Duration for medium tones (200ms).
- Note definitions can be changed to create different melodies.

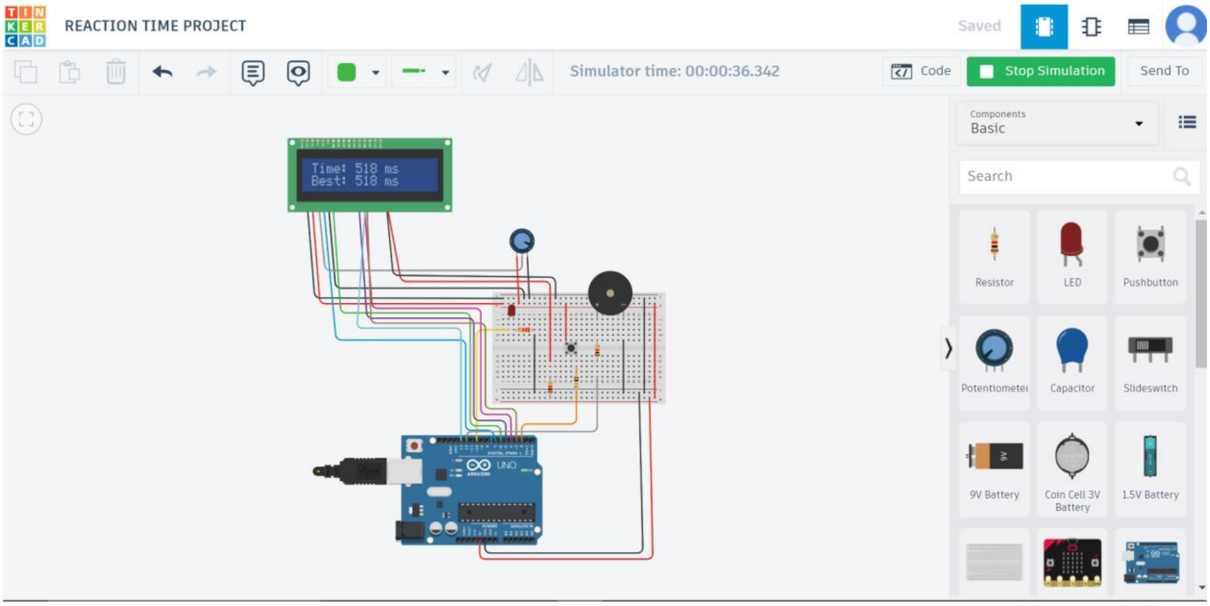
Troubleshooting:

- Inconsistent Button Response: Increase the `DEBOUNCE_TIME` value.
- Random Wait Times Too Short/Long: Adjust `MIN_WAIT_TIME` and `MAX_WAIT_TIME`.

Sound Issues:

- Ensure buzzer is connected to the correct pin (pin 12).
- Use a piezo buzzer (not a magnetic buzzer).
- Add a 100–220-ohm resistor in series with the buzzer to limit current.
- Increase tone durations if sounds are cutting off too quickly.

RESULTS AND ANALYSIS



Future Enhancements.

- Implement difficulty levels with shorter reaction windows.

- Add multi-player support to compete with friends.

- Store high scores in a JSON file for persistence after power off.

- Extend sound options with more complex melodies.

- Add volume control for sound output.

CONCLUSION

The Reaction Time Tester project demonstrates the straightforward use of microcontrollers in real-time data processing and acquisition. The project highlights the flexibility and efficiency of embedded systems by using a microcontroller to control an LED, detect user input, and display reaction times on an LCD. Microcontrollers are central to automation and control applications with inherent processing capabilities that enable direct communication with hardware devices.

This project validates the role of microcontrollers in responsive and interactive system design, especially their advantages over general-purpose microprocessors for specific, task-specific uses. Successful implementation of the Reaction Time Tester indicates the viability of employing microcontrollers to develop usable, real-world applications that require precise timing, minimal power consumption, and stability. Wireless transmission of data, more precise timing with higher-resolution timers, or input from other sensory devices for wider application are all possible directions for future extension.

REFERENCES

- www.digikey.co.il
- www.instructables.com