REPORT 1: DIGITAL LOGIC SYSTEM

GROUP 4

MCTA 3203

SEMESTER 2 2023/2024

MECHATRONICS SYSTEM INTEGRATION

DATE OF SUBMISSION:

	NAME	MATRIC NUMBER
1.	NUR SHAHEERA BINTI KAMIS	2214040
2.	NORAMIRA HUSNA BT NORKHAIRANI	2214496
3.	MUHAMMAD EIQBAL BIN HASBOLLAH	2216911

TABLE OF CONTENT

INTRODUCTION

This experiment was conducted to showcase how a common cathode 7-segment display and pushbuttons can be connected to an Arduino Uno, resulting in the creation of a digital counter. This report outlines the methodology, procedure, results, discussion, and conclusion of the experiment. By exploring the steps involved in hardware setup and software execution, the experiment aimed to provide participants with practical knowledge in circuitry, Arduino programming, and component integration. Through this endeavor, the effectiveness of the Arduino platform in facilitating experiential learning in electronics was examined.

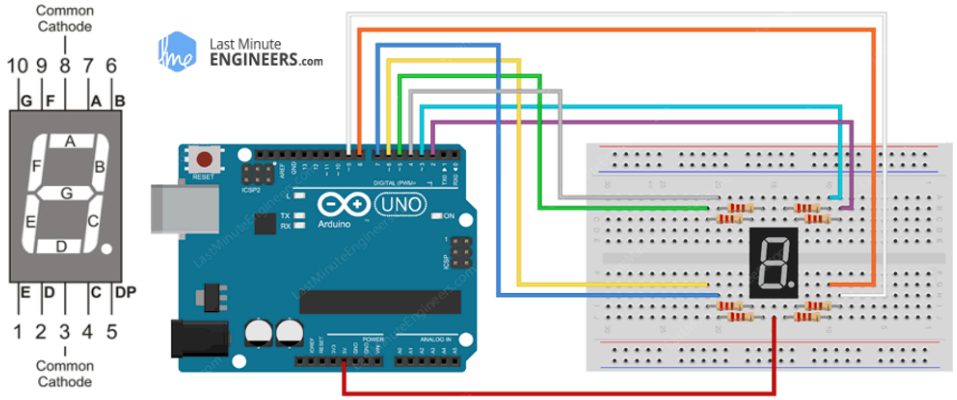
ABSTRACT

This report describes an experiment that used an Arduino Uno, pushbuttons, and a common cathode 7-segment display to demonstrate how to create a digital counter. Through the application of hardware configuration and software execution, the methodology provided practical skills in component integration, Arduino programming, and circuits. The benefit of the Arduino platform in promoting hands-on electronics learning was investigated in this experiment. The experimental setup, approach, process, findings, discussion, and conclusion are all outlined in the paper. It combines the coding concepts of the 7-segment display and matrix LED with connecting an I2C LCD to an Arduino, highlighting their easier wiring and lower pin requirements. Hardware, electrical, and software issues are covered, and problems like wiring problems and push button debounce are dealt with. Improvements for the future are also suggested.

MATERIALS AND EQUIPMENT

- Arduino Uno board
- Common cathode 7-segment display
- 220-ohm resistors (7 of them)
- Pushbuttons (2 or more)
- Jumper wires
- Breadboard

EXPERIMENTAL SETUP



METHODOLOGY

1. The common cathode 7-segment display was linked to the Arduino Uno in the following manner:

- Each of the 7 segments (a, b, c, d, e, f, g) on the display was connected to individual digital pins on the Arduino (e.g., D0 to D6).
- The common cathode pin of the display was attached to one of the GND (ground) pins on the Arduino.
- 220-ohm resistors were utilized to connect each segment pin to the Arduino pins for

current regulation.

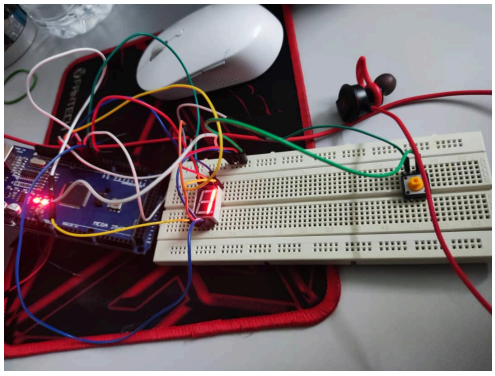
2. The pushbuttons were interfaced with the Arduino as follows:

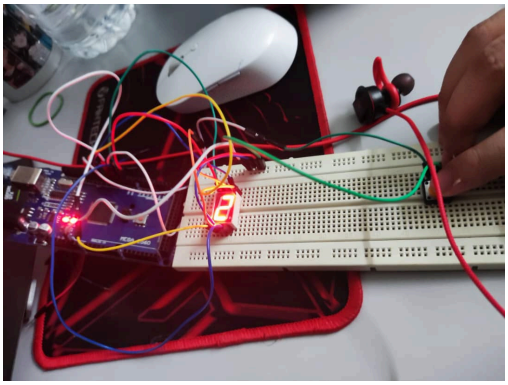
- One end of each pushbutton was connected to separate digital pins (e.g., D9 and D10), while the other end was linked to GND.
- To implement pull-up resistors for each pushbutton, one terminal of each resistor was connected to the digital pin, and the opposite terminal was connected to the 5V output of the Arduino.

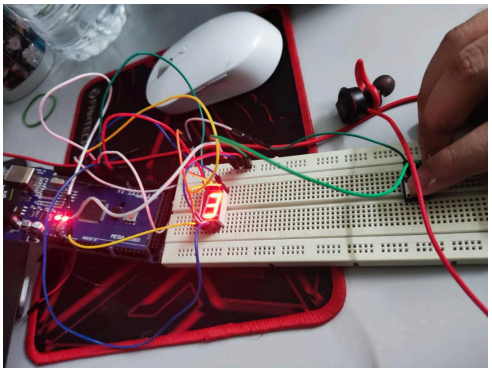
PROCEDURE

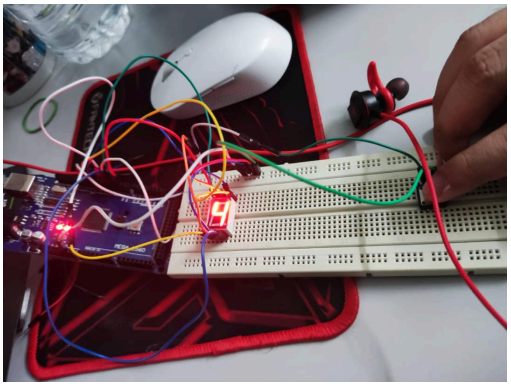
1. Built the circuit per the setup instructions provided.
2. Uploaded the provided Arduino code to your Arduino Uno.
3. Opened the Serial Monitor in the Arduino IDE.
4. Pressed the increment button to increase the count. The 7-segment display showed the numbers from 0 to 9 sequentially.
5. Press the reset button to reset the count to 0.

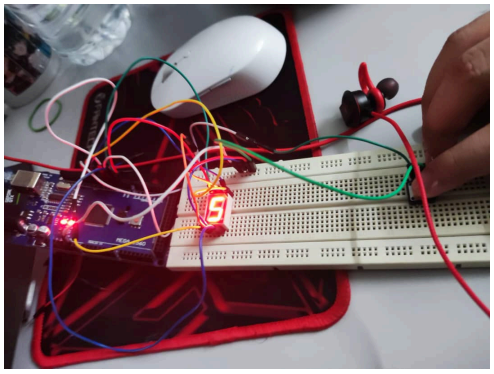
RESULTS

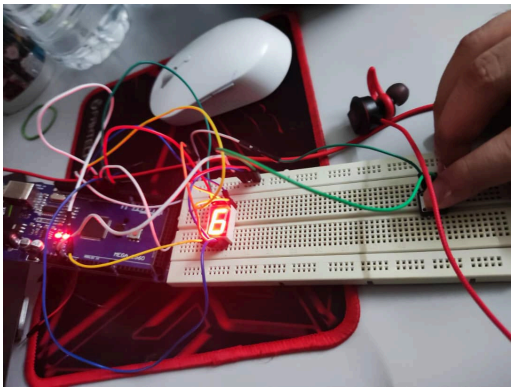


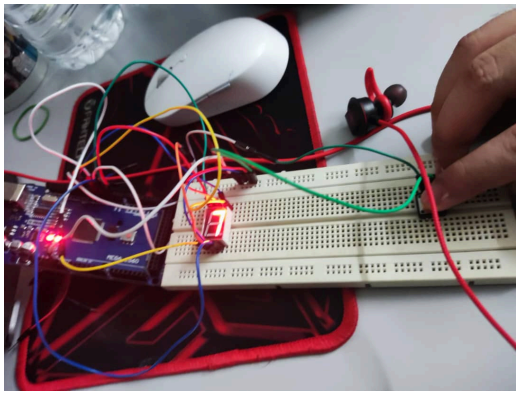


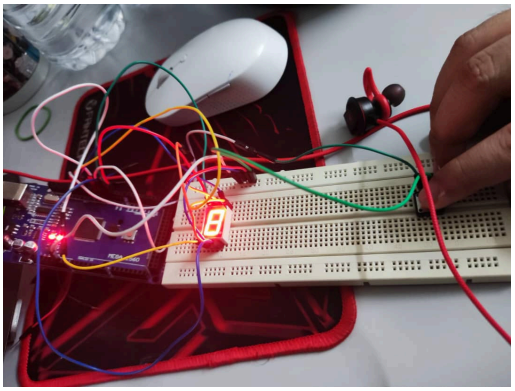


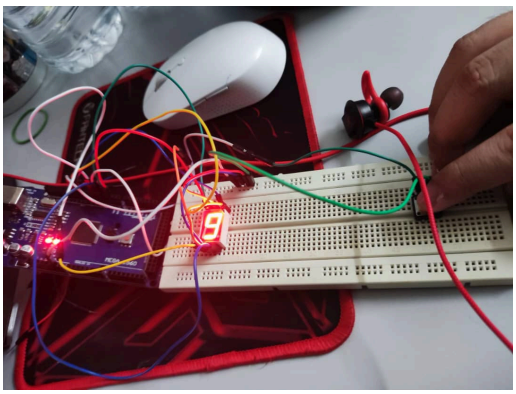


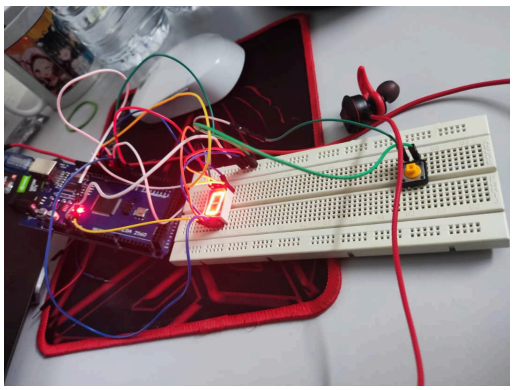












QUESTION

1) **How to interface an I2C LCD with Arduino? Explain the coding principle behind it compared with 7 segments display and matrix LED.**

Link the I2C LCD and Arduino board together. This typically includes linking the SDA pin of the LCD to the Arduino's A4 pin and the SCL pin to the Arduino's A5 pin. Verify that the appropriate power and ground pins on the Arduino are connected to the VCC and GND pins of the LCD. The 7-segment display is individually controlled by a direct connection to particular Arduino pins for the purpose of the coding principle. You must set the matching pins to HIGH or LOW depending on which segments you want to be illuminated in order to display a digit. They are set up in rows and columns for matrix LEDs, though. Setting the corresponding row pin HIGH and column pin LOW turns on a particular LED. To create the appearance of a continuous display, multiplexing is needed to rapidly cycle through the rows and columns. Meanwhile, For the 12C LCD, communication only requires two pins (SDA and SCL), regardless of the size of the display. As a result, wiring is made simpler and the Arduino requires fewer pins.

DISCUSSION

- **Hardware**

  The digital counter system is created by employing an Arduino Mega 2560 microcontroller board, a 7-segment display, and a push button. The Arduino oversees the counting operation, increasing the displayed number with every touch of the push button. The circuit configuration entails interconnecting the components using male-to-male jumper wires on a breadboard, guaranteeing accurate wiring for optimal functioning. The operating logic depends on the Arduino's digital input/output pins to connect with the push button for detecting input and the 7-segment display for displaying numbers. The challenges of push button debounce and wiring mistakes are discussed, and some solutions are provided. In addition, the project's potential for improvement is explored, including the addition of reset capability and the expansion to multiple digit counters. These enhancements would further increase the capabilities of the project and offer valuable insights into microcontroller interface, digital electronics, and embedded systems applications.

- **Electrical**

  The key electrical components utilised in this project are the 7-segment display and the push button, which are connected to the Arduino Mega 2560. The 7-segment display normally requires a current-limiting resistor for each segment to adjust the brightness and safeguard the LEDs. Furthermore, the display's common cathode/anode must be linked to the relevant pins on the Arduino to facilitate digit selection and data transfer. In order to maintain stable voltage levels while the push button is not pressed and prevent floating

inputs, it is crucial to include a pull-up or pull-down resistor. Wiring these components correctly on the breadboard, guaranteeing precise connections and resistor values, is critical for dependable operation and accurate counting. Moreover, comprehending the voltage and current prerequisites of each component guarantees compatibility and averts harm to the Arduino or the components themselves.

- **Software**

  The microcontrollers are programmed using Arduino software. Writing, building, and uploading code to Arduino boards is made easy with the help of the open-source Arduino platform. It has a text editor, a toolbar with frequently used actions, and multiple menus with access to examples, libraries, and tools. To manage the behaviour of the microcontroller, the C/C++ programming language is used, compiled, and uploaded to the Arduino board.

CONCLUSION

In conclusion, the experiment successfully demonstrated the interfacing of a common cathode 7-segment display and pushbuttons with an Arduino Uno to construct a digital counter. Through the systematic implementation of hardware connections and software programming, participants gained valuable insights into circuit building, Arduino code development, and component integration. The functional digital counter showcased the versatility and accessibility of the Arduino platform in facilitating practical learning experiences in embedded systems and electronics. By engaging in this experiment, participants were equipped with foundational skills applicable to a wide range of electronic projects and were encouraged to explore further applications and innovations in the field.

RECOMMENDATION

- Explain circuit connections and code functionality in more detail.
- Add extra features to the digital counter for a more engaging experience.
- Include troubleshooting tips for common errors.
- Gather feedback from participants to identify areas for improvement.

REFERENCES

[1]“Arduino Integrated Development Environment (IDE) v1 | Arduino Documentation | Arduino Documentation,” *Arduino.cc*, 2022. https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics/

ACKNOWLEDGEMENT

A special thanks goes out to Dr. Wahju Sediono, Dr. Ali Sophian, Dr. Zulkifli Bin Zainal Abidin, my teaching assistant, and my peers for their invaluable help and support in finishing this report. Their advice, feedback, and experience have greatly influenced the level of quality and understanding of this work. Their time, patience, and commitment to supporting my academic success are greatly appreciated.

STUDENT'S DECLARATION

Certificate of Originality and Authenticity

This is to certify that we are responsible for the work submitted in this report, that **the original work** is our own except as specified in the references and acknowledgement, and that the original work contained herein have not been untaken or done by unspecified sources or persons.

We hereby certify that this report has **not been done by only one individual** and **all of us have contributed to the report**. The length of contribution to the reports by each individual is noted within this certificate.

We also hereby certify that we have **read** and **understand** the content of the total report and that no further improvement on the reports is needed from any of the individual contributors to the report.

We, therefore, agreed unanimously that this report shall be submitted for **marking** and this **final printed report** has been **verified by us.**

Signature: 	**Read**	/
Name: Noramira Husna Bt Norkhairani	**Understand**	/
Matric No: 2214496	**Agree**	/

Signature: 	**Read**	/
Name: Nur Shaheera Bt Kamis	**Understand**	/
Matric No:2214040	**Agree**	/

Signature: 	**Read**	/
Name: Muhammad Eiqbal bin Hasbollah	**Understand**	/
Matric No: 2216911	**Agree**	/