MECHATRONICS SYSTEM INTEGRATION (MCTA 3203)

SEMESTER 1 2024/2025

WEEK 2: DIGITAL LOGIC DESIGN

SECTION 2

GROUP 8

LECTURER: ZULKIFLI BIN ZAINAL ABIDIN & WAHJU SEDIONO

NO.	GROUP MEMBERS	MATRIC NO.
1.	AHMAD DARWISH BIN AHMAD TERMIZI	2212089
2.	SYABAB ALHAQQI BIN JAAFAR	2211117
3.	WAN NAIMULLAH BIN MOHD AZRUDDIN	2214837
4.	ZAIMUDIN ZAKWAN BIN NORZAMRI	2217513
5.	ZAMIR MUTTAQIN BIN MUHAMMAD BALYAN	2212985

Date of Experiment: Wednesday, 16 October 2024

Date of Submission: Wednesday, 23 October 2024

ABSTRACT

The goal of this experiment is to explore the use of an Arduino Mega 2560 by connecting a common cathode 7-segment display to an Arduino Mega 2560 board. By doing this, we will get an understanding of the fundamentals of connecting electronic components, such as segment displays and pushbuttons, to the Arduino platform by constructing the circuit and uploading the supplied Arduino code. In this experiment we learned how to use push buttons to increase the count from 0 to 9 sequentially shown on the 7-segment display and use the reset button to reset the count to 0.

TABLE OF CONTENTS

INTRODUCTION

The Arduino Mega 2560 is a microcontroller that provides various uses such as integrating sensors, actuators and other input and output devices. They are an ideal tool for a wide range of embedded systems applications. In this experiment, we aim on using a push button to create a counter system that increments the value from 0 to 9 with each press and then resets to 0 after pressing the reset button.

The push button is a mechanical counter to input instructions to the Arduino board that will result in the output. The Arduino Mega 2560 is a microcontroller that can be programmed using the software Arduino IDE.

This experiment will help us to understand how microcontrollers manage digital inputs in real-world applications when users input their instructions. We hypothetically aim that with a single push button, the output will increase the value shown in the 7 segment by 1 and will reset back to 0 if the reset button is pressed.

MATERIALS AND EQUIPMENTS

1. **7** SEGMENT DISPLAY
2. BREADBOARD
3. ARDUINO MEGA 2560
4. PUSHBUTTON
5. MALE TO MALE JUMPER WIRE
6. RESISTOR

EXPERIMENTAL SETUP

1. For the 7 segment display, connect each of the 7 segments (a, b, c, d, e, f, g) of the display to separate digital pins on the Arduino.
2. Then, connect the common anode pin of the display to the 3.3v pin on the Arduino.
3. For the pushbutton, connect one from the legs to a digital pin on the Arduino, another one to the ground (GND) pin and another one to the 5v pin through a resistor as shown in Figure 1.

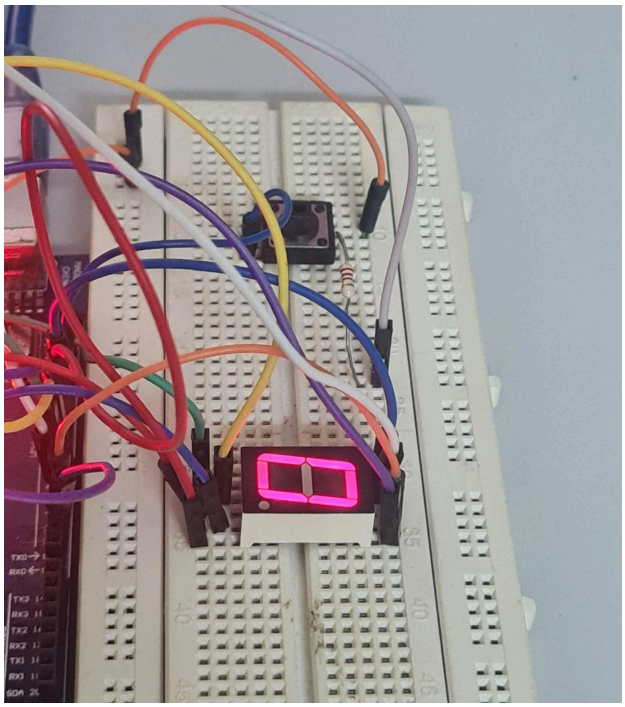


Figure 1: Hardware setup

METHODOLOGY

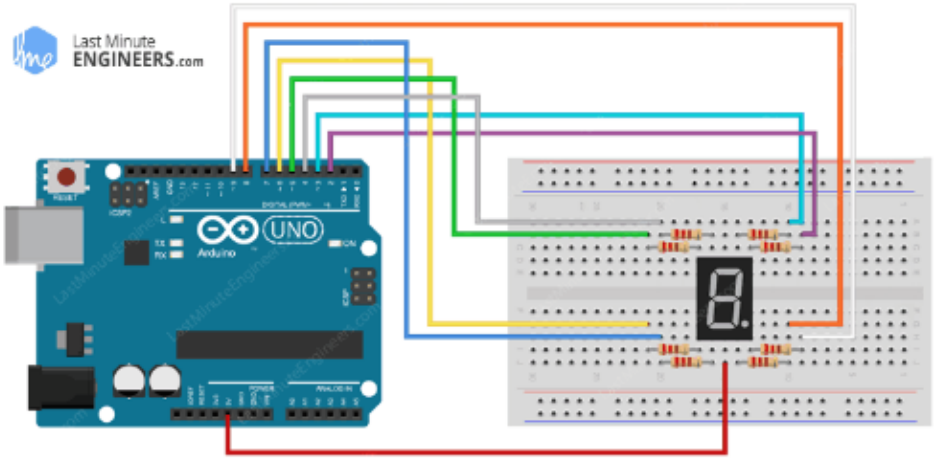
1. Setup the Arduino Mega 2560
2. Code implementation
3. Testing
4. Code snippet

```
// Define the pins for each segment (D0 to D6)
const int segmentA = 3; // D0
const int segmentB = 2; // D1
const int segmentC = 8; // D2
const int segmentD = 7; // D3
const int segmentE = 6; // D4
const int segmentF = 4; // D5
const int segmentG = 10; // D6
int i = 0; //as counter

void setup() {
  // Initialize the digital pins as OUTPUTs
  pinMode(segmentA, OUTPUT);
  pinMode(segmentB, OUTPUT);
  pinMode(segmentC, OUTPUT);
  pinMode(segmentD, OUTPUT);
  pinMode(segmentE, OUTPUT);
  pinMode(segmentF, OUTPUT);
  pinMode(segmentG, OUTPUT);
  pinMode(12, INPUT); //BUTTON
  Serial.begin(9600);
}
/*
 0 = A,B,C,D,E,F
 1 = B,C
 2 = A,B,G,E,D
 3 = A,B,C,D,G
 4 = F,G,B,C
 5 = A,F,G,C,D
 6 = A,F,G,E,D,C
 7 = A,B,C
 8 = A,B,C,D,E,F,G
 9 = A,B,C,D,F,G
 */
void loop() {
   if(i==0){
// turn on the display according to the counter
 //0
  digitalWrite(segmentA,  LOW);
  digitalWrite(segmentB,  LOW);
  digitalWrite(segmentC,  LOW);
  digitalWrite(segmentD,  LOW);
  digitalWrite(segmentE,  LOW);
  digitalWrite(segmentF,  LOW);
  digitalWrite(segmentG,  HIGH);

 }
 else if(i==1){
  //1
  digitalWrite(segmentB,  LOW);
  digitalWrite(segmentC,  LOW);
  digitalWrite(segmentA,  HIGH);
  digitalWrite(segmentD,  HIGH);
  digitalWrite(segmentE,  HIGH);
  digitalWrite(segmentF,  HIGH);
  digitalWrite(segmentG,  HIGH);


 }
 else if(i==2){
  //2
  digitalWrite(segmentA,  LOW);
  digitalWrite(segmentB,  LOW);
  digitalWrite(segmentG,  LOW);
  digitalWrite(segmentE,  LOW);
  digitalWrite(segmentD,  LOW);
  digitalWrite(segmentF,  HIGH);
  digitalWrite(segmentC,  HIGH);

 }
 else if(i==3){
  //3
  digitalWrite(segmentA,  LOW);
  digitalWrite(segmentB,  LOW);
  digitalWrite(segmentC,  LOW);
  digitalWrite(segmentD,  LOW);
  digitalWrite(segmentG,  LOW);
  digitalWrite(segmentE,  HIGH);
```

```
 digitalWrite(segmentF,  HIGH);

}
else if(i==4){
 //4
 digitalWrite(segmentF,  LOW);
 digitalWrite(segmentG,  LOW);
 digitalWrite(segmentB,  LOW);
 digitalWrite(segmentC,  LOW);
 digitalWrite(segmentA,  HIGH);
 digitalWrite(segmentE,  HIGH);
 digitalWrite(segmentD,  HIGH);

}
else if(i==5){
 //5
 digitalWrite(segmentA,  LOW);
 digitalWrite(segmentF,  LOW);
 digitalWrite(segmentG,  LOW);
 digitalWrite(segmentC,  LOW);
 digitalWrite(segmentD,  LOW);
 digitalWrite(segmentB,  HIGH);
 digitalWrite(segmentE,  HIGH);

}
else if(i==6){
 // 6
 digitalWrite(segmentA,  LOW);
 digitalWrite(segmentF,  LOW);
 digitalWrite(segmentG,  LOW);
 digitalWrite(segmentC,  LOW);
 digitalWrite(segmentD,  LOW);
 digitalWrite(segmentE,  LOW);
 digitalWrite(segmentB,  HIGH);

}
else if(i==7){
 // 7
 digitalWrite(segmentA,  LOW);
 digitalWrite(segmentC,  LOW);
 digitalWrite(segmentB,  LOW);
 digitalWrite(segmentD,  HIGH);
 digitalWrite(segmentE,  HIGH);
 digitalWrite(segmentF,  HIGH);
 digitalWrite(segmentG,  HIGH);

}
else if(i==8){
 // 8
 digitalWrite(segmentA,  LOW);
 digitalWrite(segmentB,  LOW);
 digitalWrite(segmentC,  LOW);
 digitalWrite(segmentD,  LOW);
 digitalWrite(segmentE,  LOW);
 digitalWrite(segmentF,  LOW);
 digitalWrite(segmentG,  LOW);

}
else if(i==9){
 // 9
 digitalWrite(segmentA,  LOW);
 digitalWrite(segmentB,  LOW);
 digitalWrite(segmentC,  LOW);
 digitalWrite(segmentD,  LOW);
 digitalWrite(segmentF,  LOW);
 digitalWrite(segmentG,  LOW);
 digitalWrite(segmentE,  HIGH);

}

 if(digitalRead(12)==1)
 {
   while(digitalRead(12)==1)
   {
    Serial.println(i);
   }
   i++; //increase counter with each push
button
 }
   //start count back to 0
 if(i == 10)
 {
   i = 0;
 }
 delay(1000);
}
```

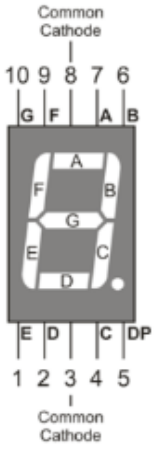
DATA COLLECTION

i) Figure: Circuit Diagram
Figure shows how the components were connected.

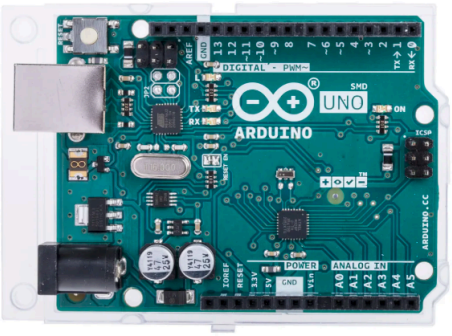


ii) Instruments

a. Common cathode 7-segment display



b. Arduino Mega 2560



c. Resistors

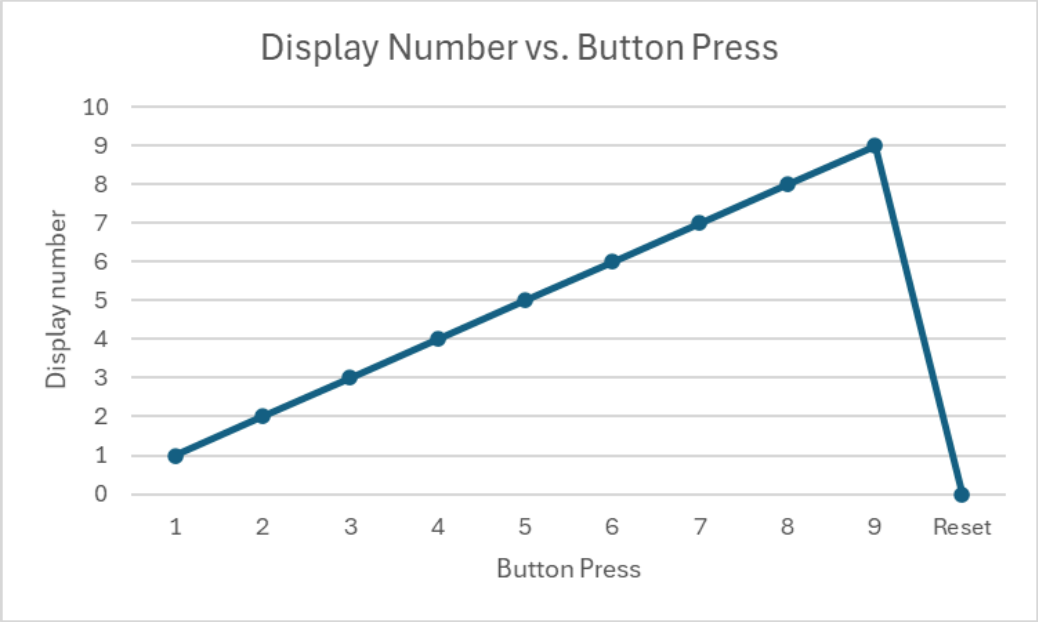
d. Pushbutton

e. Jumper wires
f. Breadboard

Table 1 of expected and actual display output of the 7-segment display with the time interval.

Button Press	Expected Display Output	Actual Display Output	Time Interval (ms)
Increment Button (1st)	1	1	500
Increment Button (2nd)	2	2	1000
Increment Button (3rd)	3	3	1600
Increment Button (4th)	4	4	2000
Increment Button (5th)	5	5	2500
Increment Button (6th)	6	6	3000
Increment Button (7th)	7	7	3500
Increment Button (8th)	8	8	4000
Increment Button (9th)	9	9	4500
Reset Button Press	0	0	2500

Graph of button press vs display number was plotted.



The graph shows how the number on the 7-segment display increases sequentially with each press of the increment button, and resets when the reset button is pressed.

DATA ANALYSIS

Based on the data collected, with each press of the increment button, the number on the display increased sequentially from 0 to 9. The time interval between button presses remained consistent at 500 milliseconds, reflecting the delay programmed in the Arduino code. When the reset button was pressed, the display reset to 0. The system functioned as intended, as there were no inconsistencies observed between the expected and actual display outputs. The steady timing and accurate display progression suggest that both the button inputs and the 7-segment display interfacing with the Arduino were reliable and responsive throughout the experiment.

RESULT

The experiment was successful in obtaining its goal. When the increment button was pressed, the 7-segment display successfully displayed the numbers from 0 to 9 then reset to 0. The actual output matched with the expected output as shown at Table 1.

DISCUSSION

The experiment successfully demonstrated using Arduino MEGA2560 to control a 7-segment display using a pushbutton. The actual results matched the expected outcomes, showing that the system was properly designed. The time intervals between button presses and related display changes were constant, confirming the reliability of the circuit design.

Some potential sources of errors can to include:

- Resistor Values: Incorrect resistor values could cause inaccurate voltage drops, potentially affecting the display brightness or causing faulty outputs, but this was not observed.
- Button Debouncing: If the button wasn't debounced correctly, several counts might have been registered with a single press. However, this was not observed in the results, indicating that either debouncing was inherently addressed or the delay was sufficient to avoid this issue.

CONCLUSION

The push-button counter worked as intended, accurately counting from 0 to 9. The experiment demonstrated the importance of digital input systems to ensure a reliable user interaction. The results aligned with the expected hypothesis, confirming that the code effectively gives the intended output.

RECOMMENDATIONS

We recommend implementing a long-press functionality for the button such as reset counter on long press. Additionally, make another button to decrease the value shown on the 7 segment display.

REFERENCES

- https://www.arduino.cc/en/Guide/Introduction
- https://lastminuteengineers.com/seven-segment-arduino-tutorial/

ACKNOWLEDGEMENTS

Special thanks to ZULKIFLI BIN ZAINAL ABIDIN & WAHJU SEDIONO for their guidance and support during this experiment.

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 no further improvement on the reports is needed from any of the individual's 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: *darwish* Read [/]
Name: AHMAD DARWISH BIN AHMAD TERMIZI Understand [/]
Matric Number: 2212089 Agree [/]
Contribution: Data Collection, Data Analysis

Signature: *syabab* Read [/]
Name: SYABAB ALHAQQI BIN JAAFAR Understand [/]
Matric Number: 2211117 Agree [/]
Contribution: Abstract, Introduction, Conclusion

Signature: *naim* Read [/]
Name: WAN NAIMULLAH BIN MOHD AZRUDDIN Understand [/]
Matric Number: 2214837 Agree [/]
Contribution: Materials and Equipments, Recommendations

Signature: *zaim* Read [/]
Name: ZAIMUDIN ZAKWAN BIN NORZAMRI Understand [/]
Matric Number: 2217513 Agree [/]
Contribution: Result, Discussion

Signature: *zamir* Read [/]
Name: ZAMIR MUTTAQIN BIN MUHAMMAD BALYAN Understand [/]
Matric Number: 2212985 Agree [/]
Contribution: Experimental setup, Methodology