

Hardware Assignment

Digital Clock

EE1003

Dasari Manikanta
(EE24BTECH11013)

Problem Statement: Make a digital clock using seven-segment displays in AVR-GCC, without using flip-flops.

ABSTRACT

This report details the design and implementation of a digital clock using an Arduino Uno, six seven-segment displays, a 7447 IC, 270-ohm resistors, and a breadboard. The project demonstrates the integration of hardware components to create a functional digital clock. The report explains the role of each component, the procedure for assembling the clock, and the significance of such projects in daily life.

1 INTRODUCTION

Digital clocks are ubiquitous in modern life, found in devices ranging from microwaves to smartphones. This project aims to build a simple digital clock using basic electronic components. The clock displays hours and minutes using six seven-segment displays controlled by an Arduino Uno.

2 COMPONENTS USED

2.1 *Arduino Uno*

The Arduino Uno is a microcontroller board based on the ATmega328P. It serves as the brain of the clock, controlling the seven-segment displays and managing timekeeping.

2.2 *Breadboard*

A breadboard is used to create temporary electrical connections between components. It allows for easy prototyping without soldering.

2.3 *Jumper Cables*

Jumper cables are used to connect components on the breadboard to the Arduino Uno. They provide a flexible and reusable way to make electrical connections.

2.4 Six Seven-Segment Displays

Seven-segment displays are used to display numeric digits. Each display consists of seven LEDs arranged in a specific pattern to represent numbers from 0 to 9. Six displays are used to show hours (two digits), minutes (two digits), and seconds (two digits).

2.5 7447 IC

The 7447 is a BCD-to-seven-segment decoder/driver IC. It converts binary-coded decimal (BCD) input from the Arduino into signals that drive the seven-segment displays.

2.6 Resistors

Six 270-ohm resistors are used to limit the current flowing through the LEDs in the seven-segment displays, preventing them from burning out.

3 ROLE OF COMPONENTS IN DAILY LIFE

3.1 Arduino Uno

Arduino boards are widely used in prototyping and DIY projects. They are used in home automation, robotics, and educational projects.

3.2 Breadboard

Breadboards are essential for testing and prototyping electronic circuits. They are used in educational settings and by hobbyists.

3.3 Jumper Cables

Jumper cables are used in various electronic projects to connect components. They are reusable and make circuit assembly easier.

3.4 Seven-Segment Displays

Seven-segment displays are used in digital clocks, calculators, and other devices that display numeric information.

3.5 7447 IC

The 7447 IC is used in devices that require numeric displays, such as digital clocks and counters.

3.6 Resistors

Resistors are fundamental components in electronics, used to control current and voltage levels in circuits. The 270-ohm resistors ensure the LEDs in the seven-segment displays operate within safe current limits.

4 PROCEDURE

4.1 Step 1: Setting Up the Breadboard

- 1) Place the six seven-segment displays on the breadboard.
- 2) Connect the common cathode (or anode) of each display to the ground (or VCC) using jumper cables.

4.2 Step 2: Connecting the 7447 IC

- 1) Place the 7447 IC on the breadboard.
- 2) Connect the BCD input pins (A, B, C, D) of the 7447 IC to the digital output pins of the Arduino Uno (e.g., pins 2, 3, 4, 5).
- 3) Connect the output pins (a, b, c, d, e, f, g) of the 7447 IC to the corresponding segments of the seven-segment displays.

4.3 Step 3: Adding Resistors

- 1) Connect a 270-ohm resistor between each segment of the seven-segment displays and the output pins of the 7447 IC.
- 2) This limits the current through the LEDs, protecting them from damage.

4.4 Step 4: Programming the Arduino Uno

- 1) Write a program in the Arduino IDE to control the seven-segment displays.
- 2) The program should increment the time every second and update the displays accordingly.
- 3) Use the 'millis()' function for accurate timekeeping without blocking the code.
- 4) Upload the program to the Arduino Uno.

4.5 Step 5: Testing the Clock

- 1) Power the Arduino Uno using a USB cable or an external power supply.
- 2) Verify that the seven-segment displays show the correct time (HH:MM:SS).
- 3) Adjust the program if necessary to correct any errors.

5 CONCLUSION

This project successfully demonstrates the construction of a digital clock using an Arduino Uno, six seven-segment displays, and other electronic components. The project highlights the importance of understanding basic electronics and programming in creating functional devices. The use of 270-ohm resistors ensures the longevity of the LEDs, while the 7447 IC simplifies the control of the seven-segment displays.