A PROJECT REPORT ON Led effects using Arduino and 74HC595 Shift Registers.

Crimson Technical Collage

Butwal-11, Rupandehi, Nepal

Diploma in Computer Engineering 2021 Batch

DEPARTMENT OF COMPUTER ENGENNERING

A Major Project Report Submitted in the Partial Fulfillment of requirement of the Diploma Degree in Computer Engineering Awarded by CTEVT.

SUBMITED BY D-COM 3th SEMETER STUDENTS

MEMBERS	REGISTRATION NO
MANISH SINJALI	DCOM-00315-021
MANOJ SHRESTHA	DCOM-00320-021
KRISHNA ARAYL	DCOM-00322-021
LOK MAYA KUMAR	DCOM-00313-021

Date: 2022 / 06 / 18

Crimson Technical College

Butwal-11, Rupendehi, Nepal

……. ……..……

Er. Rajiv Aryal **External Examiner**

Project Supervisor

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all those who have supported and guided me throughout the development of this project, **"Led effects using Arduino and 74HC595 Shift Registers."**

First and foremost, I extend my heartfelt thanks to my project guide, Er. Rajiv Aryal, for his invaluable advice, constant encouragement, and expertise, which were instrumental in the successful completion of this project. His insightful feedback and guidance helped me navigate challenges and improve the overall quality of the work.

I am deeply grateful to my group members for their unwavering support, patience, hard work and encouragement during the course of this project.

Lastly, I would like to acknowledge the wealth of knowledge provided by the various online communities and open-source platforms that helped me overcome technical challenges. Without the support of these communities, this project would not have been possible.

Thank you all for your contributions and encouragement.

- **Project Leader**

- MANISH SINJALI

Table of Contents

1. Introduction

In this project, we will control 16 LEDs using an Arduino and two 74HC595 shift registers. The shift registers allow us to expand the number of output pins using only three pins from the Arduino. Various lighting effects will be implemented to demonstrate the versatility of shift registers.

2. Components Required

- Arduino (Uno/Nano/Mega)
- 74HC595 Shift Register (2 pieces)
- 16 LEDs
- 16 Current Limiting Resistors (220Ω - 330Ω)
- Jumper Wires
- Breadboard
- Power Supply (USB or external 5V source)

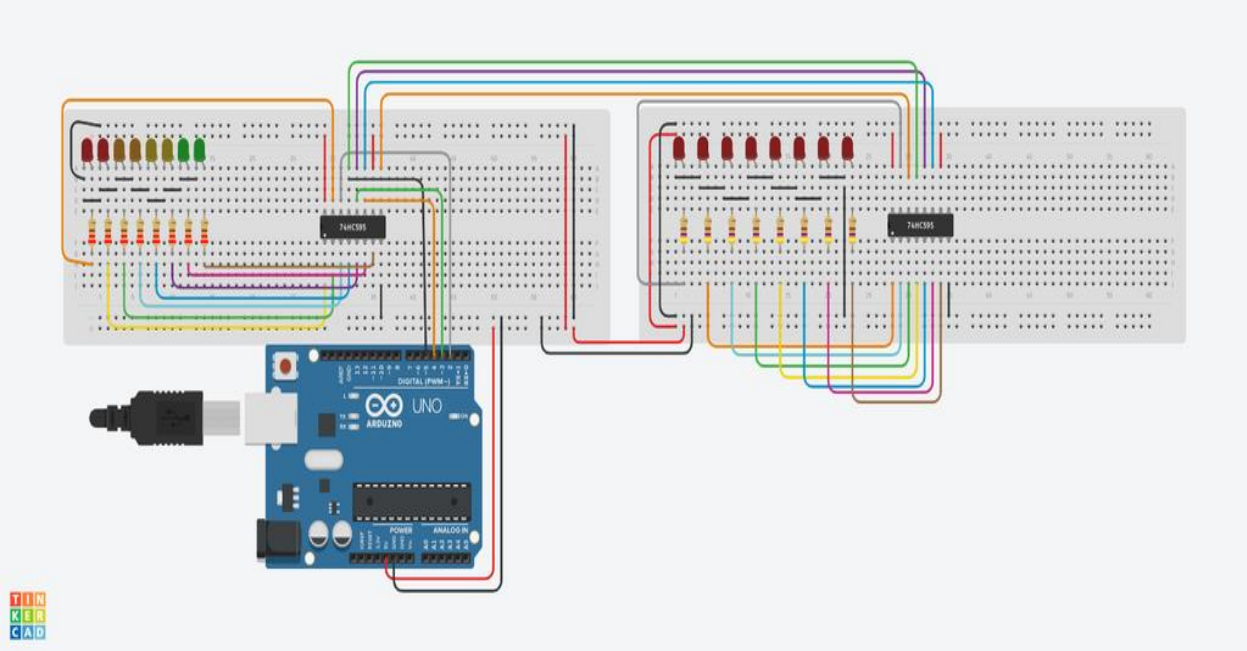
3. Understanding 74HC595 Shift Register

The 74HC595 is an 8-bit shift register that extends the number of output pins available on an Arduino. The three main pins used to control the shift register are:

- **Data (DS)**: Serial data input
- **Clock (SH_CP)**: Shifts data on each rising edge
- **Latch (ST_CP)**: Updates the output with stored data

By cascading two 74HC595 ICs, we can control 16 LEDs using only three Arduino pins.

4. Circuit Diagram



To connect the components:

1. Connect **VCC** and **GND** of both 74HC595 to 5V and GND of the Arduino.
2. Connect **DS**, **SH_CP**, and **ST_CP** of the first 74HC595 to Arduino digital pins.
3. Cascade the two shift registers by connecting the **Q7' (Serial Out)** of the first 74HC595 to the **DS** of the second.
4. Connect LED anodes to the **Q0-Q7** outputs of each shift register, with resistors in series.
5. Connect LED cathodes to GND.

5. Code Explanation

The Arduino code sends data to the shift registers in a sequence to create various LED effects.

We use the **shiftOut()** function to send 8-bit data serially.

Sample Code:

```
 // Define shift register control pins
const int dataPin = 2;   // Data input to shift register
const int clockPin = 3;  // Clock pin for data shifting
const int latchPin = 4;  // Latch pin to update output


void setup() {
  // Set pin modes as outputs
  pinMode(dataPin, OUTPUT);
  pinMode(clockPin, OUTPUT);
  pinMode(latchPin, OUTPUT);
}


void loop() {
  // Loop through each LED and turn it on sequentially
  for (int i = 0; i < 16; i++) {
    shiftOutData(1 << i); // Shift bit left to turn on one LED at a time
    delay(100); // Small delay for visibility
```

```
  }
}

// Function to send 16-bit data to the shift registers
void shiftOutData(uint16_t data) {
  digitalWrite(latchPin, LOW); // Prepare for data transfer

  // Send the higher 8 bits to the first shift register
  shiftOut(dataPin, clockPin, MSBFIRST, data >> 8);

  // Send the lower 8 bits to the second shift register
  shiftOut(dataPin, clockPin, MSBFIRST, data & 0xFF);

  digitalWrite(latchPin, HIGH); // Latch the data to update the outputs
}
```

6. Implementation Steps

1. Assemble the circuit on a breadboard as per the diagram.
2. Upload the code to the Arduino.
3. Observe the LED patterns and adjust delays or data sequences to modify effects.

7. LED Effects and Description

1. **Chasing Effect** - LEDs light up sequentially.
2. **Reverse Chasing** - LEDs light up in reverse order.
3. **Ping Pong** - Light moves back and forth.
4. **Alternating Blink** - Even and odd LEDs blink alternately.
5. **Running Pattern** - Multiple LEDs move in a wave-like pattern.
6. **Random Twinkle** - LEDs turn on and off randomly.
7. **Fading Effect** - Dimming effect using PWM (requires modification).
8. **Center Expansion** - LEDs light from the center outward.
9. **Edge to Center** - LEDs light from edges toward the center.
10. **Flashing All** - All LEDs blink together.
11. **Knight Rider Effect** - LED movement similar to the Knight Rider car lights.
12. **Binary Counting** - LEDs display a binary count.
13. **Ripple Effect** - Creates a ripple-like motion.
14. **Wave Effect** - Smooth lighting transition across LEDs.
15. **Expanding and Contracting Wave** - A more complex version of the wave effect.
16. **Customizable Patterns** - Modify the code to create new effects.

8. Troubleshooting

- **No LEDs lighting up?** Check wiring and ensure proper power supply.
- **Wrong LED sequence?** Verify the order of data shifting.
- **Unstable behavior?** Use proper resistors and avoid loose connections.

9. Conclusion

This project demonstrates how to control multiple LEDs efficiently using shift registers. By modifying the code, various lighting effects can be achieved, making it a great learning experience for beginners in Arduino and embedded systems.

Future Enhancements:

- Implement fading effects using PWM.
- Add user-controlled button inputs for effect selection.
- Integrate more shift registers for additional LEDs.