**DSM Lab Report**

**Name:** Anirudh Kaushik

**Roll No:** 2020111015

**Group Number:** 5

**Experiment Part A**

**Objective**

To build a circuit for a decade counter understand its usage. In this experiment, you need to make a sequential circuit that counts from 0 to 9 and then resets back to 0 and so on.

**Theory**

A decade counter also known as a binary-coded decimal is a serial digital counter that counts ten digits. It resets for every new clock input. It goes through 10 unique combinations of output. A decade counter counts in a sequence of ten and then returns back to zero after the count of nine. Obviously, to count up to a binary value of nine, the counter must have at least four flip-flops within its chain to represent each decimal digit.

**555 timer:** 555 timer IC is an integrated circuit (chip) used in a variety of timer, delay, pulse generation, and oscillator applications. This IC is useful for generating accurate time delays and oscillations. It is mainly useful for generating non-sinusoidal waveforms like square, ramp, pulse etc.

|  |
| --- |
|  |

**IC 74HC93:** 74HC93 are 4-bit binary ripple counters. In a 4-bit ripple counter, the output Q0 must be connected externally to input CP1. The input count pulses are applied to clock input CP0. Simultaneous frequency divisions of 2, 4, 8 and 16 are performed at the Q0, Q1, Q2 and Q3 outputs. As a 3-bit ripple counter, the input count pulses are applied to input CP1. Simultaneous frequency divisions of 2, 4 and 8 are available at the Q1, Q2 and Q3 outputs.

**IC CD4511:** IC CD4511 is a BCD to 7-segment latch decoder driver IC. This IC is used where we need to driving common-cathode displays like 7-segment display, low voltage fluorescent display, and incandescent display. It has high output-current-sourcing up to 25mA comes with a lamp test and blanking capability to test the display. It is having a DC supply input that ranges from 3 to 18V.

**Experiment setup/ procedure**

**Materials required:**

|  |  |
| --- | --- |
| 1 | Timer |
| 8 | 1 kΩ Resistor |
| 1 | Voltage Multimeter |
| 1 | 5 , 5 Power Supply |
| 1 | 4-Bit Binary Counter |
| 1 | 7-Segment Decoder |
| 1 | 470 uF Capacitor |
| 1 | Cathode 7 Segment Display |
| 4 | Red LED |

**Procedure:**

Assemble the circuit as shown in the block diagram given below and verify its functioning.

**Block diagram:**

|  |
| --- |
|  |

**Code:**

Not used.

**Observations**

The decade counter counts from 0 to 9 and then resets back to 0 and then resumes counting.

**Conclusion**

Successfully created a decade counter using 7 segment display, 555 timer IC, 4-bit binary counter and 7 segment decoder IC and verified its functioning.

**Tinkercad link with circuit**

**Circuit Diagram**

|  |
| --- |
|  |

**Experiment Part B**

**Objective**

To build a circuit for shift register and understand its usage.

**Theory**

At some time or another, you may run out of pins on your Arduino board and need to extend it with shift registers. This experiment is based on the 74HC595. The datasheet refers to the 74HC595 as an "8-bit serial-in, a serial or parallel-out shift register with output latches; 3-state." In other words, you can use it to control 8 outputs at a time while only taking up a few pins on your microcontroller.

**Experiment setup/ procedure**

**Materials required:**

|  |  |
| --- | --- |
| 1 | Arduino Uno R3 |
| 1 | 8-Bit Shift Register |
| 8 | 1 kΩ Resistor |
| 8 | Red LED |

**Procedure:**

Write the code to count from 0 to 255 and glow the 8 LEDs in order.

**Code:**

int latchPin = 13;

//Pin connected to SH\_CP of 74HC595

int clockPin = 12;

////Pin connected to DS of 74HC595

int dataPin = 11;

void setup() {

//set pins to output because they are addressed in the main loop

pinMode(latchPin, OUTPUT);

pinMode(clockPin, OUTPUT);

pinMode(dataPin, OUTPUT);

}

void loop() {

//count up routine

for (int val = 0; val < 256; val++) {

//ground latchPin and hold low for as long as you are transmitting

digitalWrite(latchPin, LOW);

shiftOut(dataPin, clockPin, MSBFIRST, val);

//return the latch pin high to signal chip that it

//no longer needs to listen for information

digitalWrite(latchPin, HIGH);

delay(200);

}

}

**Observations**

The counter counts from 0 to 255, delay is 200ms i.e. 1unit of time = 200ms.

**Conclusion**

Successfully created an 8-bit counter using shift register logic.

**Tinkercad Link with Circuit**

**Circuit Diagram**

|  |
| --- |
|  |

**Experiment Part B.2**

**Objective**

Using the circuit made above write the code to take input from the user (range 0-7) and glow the corresponding LED.

**Experiment setup/ procedure**

**Materials required:**

|  |  |
| --- | --- |
| 1 | Arduino Uno R3 |
| 1 | 8-Bit Shift Register |
| 8 | 1 kΩ Resistor |
| 8 | Red LED |

**Procedure:**

Use the previous circuit and write the required Arduino code to make the corresponding LED glow.

**Code:**

int latchPin = 13;

//Pin connected to SH\_CP of 74HC595

int clockPin = 12;

////Pin connected to DS of 74HC595

int dataPin = 11;

void setup() {

//set pins to output because they are addressed in the main loop

pinMode(latchPin, OUTPUT);

pinMode(clockPin, OUTPUT);

pinMode(dataPin, OUTPUT);

Serial.begin(9600);

}

int val;

int A;

void loop()

{

if(Serial.available() > 0)

{

while(!Serial.available())

{}

A = Serial.read(); // x would be an integer between 0 and 255

// depending on the ascii value of the character read

A = A - '0'; // Subtracting ascii value of 0 from x.

val = (1 << A);

digitalWrite(latchPin, LOW);

shiftOut(dataPin, clockPin, MSBFIRST, val);

//return the latch pin high to signal chip that it

//no longer needs to listen for information

digitalWrite(latchPin, HIGH);

Serial.print("A = ");

Serial.println(A);

Serial.print("val = ");

Serial.println(val);

}

}

**Observation**

The circuit glows LED 0 when 0 is entered, LED 1 for input 1 and so on till input 7.

**Conclusion**

Successfully made a circuit to make the corresponding LED glow using shift register logic.

**Tinkercad Link with Circuit Part B.2**

**Circuit Diagram**

|  |
| --- |
|  |