**-DEPARTMENT OF SOFTWARE ENGINEERING-**

**-FAST NUCES ISLAMABAD-**



**BDS-3A**

**DLD Lab**

**Project**

**Lab Project (Counter via D Flip Flop)**

**Submitted to:**

Ms. Nabeelah Maryam

**Submitted by:**

**Name**: Muhammad Noor / Tayyab Hamad / Ahmed Tanweer

**Registration** **No:** 23i-2520 / 23i-0053 / 23i-3019

**Class / Section:** Ds-A

**Date: 29/11/2024**

**Task 1:   
Objectives:**

Main objective of our Project is to design a Binary (Counter) from 0-15 using D-Flip Flops.   
 **Significance:**

Counters have utmost significance in many of our real-world applications, although most of those are decimal counters, binary counters too although have much importance but mostly in the digital world, e.g.:

1. Operating Systems

CPUs while multitasking need binary timers (which are essentially the same circuit with just the main trigger to be a timer instead of a button) to keep track of time to switch between tasks frequently.

1. Clock Division

Used in frequence division to generate slower clock signals from a faster source.

1. Adress Sequencing in Memory Units:

In RAMs or ROMs, binary counters are used to generate addresses for reading or writing data sequentially.

These are a few of multiple examples of how ironically in such an advanced world, things as small as a binary counter still hold utmost significance.

**Functionality:**

The project circuit essentially counts from 0 – 15, incrementing value by a 1 every time the button is pressed. Additionally, there is a pause button that when pressed, stops the circuit and the circuit doesn’t count forward even with the button being pressed leaving us with the reset button, which as the name implies resets the counter back to 0.

**Task 2:**

**Main Design and Logic Used:**

1.The circuit uses 4 D- FFs to construct a 4-BIT binary counter, with each flip flop representing one bit in the binary counter.

2.Flip Flops are arranged in a sequence , where the clock entry of each flip flop is controlled by the Q output of previous flip flop. In this manner, out setup allows the counter to increment in a sequence from 0-15.

3.Each time a falling edge (1-0) of clock occurs, the flip flop implemented toggle its state as the inverted Q is connected to the D input of the flip flop connected to the right. This causes it to toggle every clock cycle. Q0 changes state with each clock pulse. Q1 toggles only when Q0 transitions from (1-0), Q2 toggles only when Q1 transitions from (1-0), Q3 toggles only when Q2 transitions from (1-0). Each flip flop plays its part in this ripple effect setup.

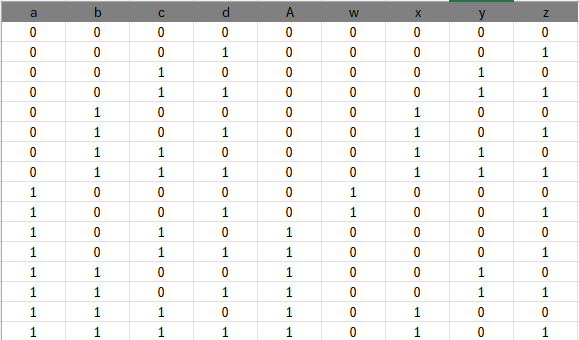
4.As the count reaches (1111) – 15, the clock pulse causes all flip-flops connected to reset (0000)-0. This reset is triggered by the falling edge of the clock input. This step will ensure that when the counter reaches (1111) -15, it will return to (0000) – 0, and counting will be repeated like a cycle.

5.Reset button relates to Reset Pins of all flip-flops and will reset counter to (0000)-0 when pressed.

6.Pause button interrupts the clock signal using an AND gate, halting the counter when pressed. The counter will remain at its current position until the pause button is again released from 1-0.

7.Segment Display is also connected to decode the 4-bit output from the flip-flops and display the 7-segment display. The current state of the flip-flop will be converted to its equivalent and will be displayed using 7-segment display.

**Truth Table And K-Map:**

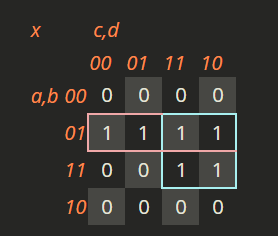
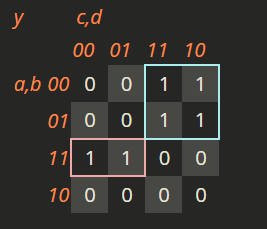
****

**A screenshot of a computer

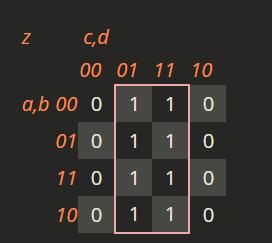
Description automatically generated A screenshot of a computer

Description automatically generated**

**A = ab + ac**  **w = a**

** **

**x = b + bc y = ab + c**

****

**z=d**

**Implementation Process:**

1.Firstly, place 4D FFs to store 4 bits (Q0 to Q3).

2. Connect Q0 to clock of Q1, Q1 to Q2, Q2 to Q3 to form a ripple counter which will count the bits.

3.Then, connect D input to inverted Q of each flip flop to toggle on each pulse.

4.Then use an AND gate to block the clock pulse to Q0 when pressed, so that it halts the counting process.

5.Connect the reset pins of all flip-flops to reset the counter to 0000 when pressed.

6.When the counter reaches 10, we use our derived logic to set the right 7 segment display to 0 and the left 7 segment display to 1. In this way, our counter displays the correct decimal equivalent on the screen.

**Task3:**

**Components List:**

**Software:**

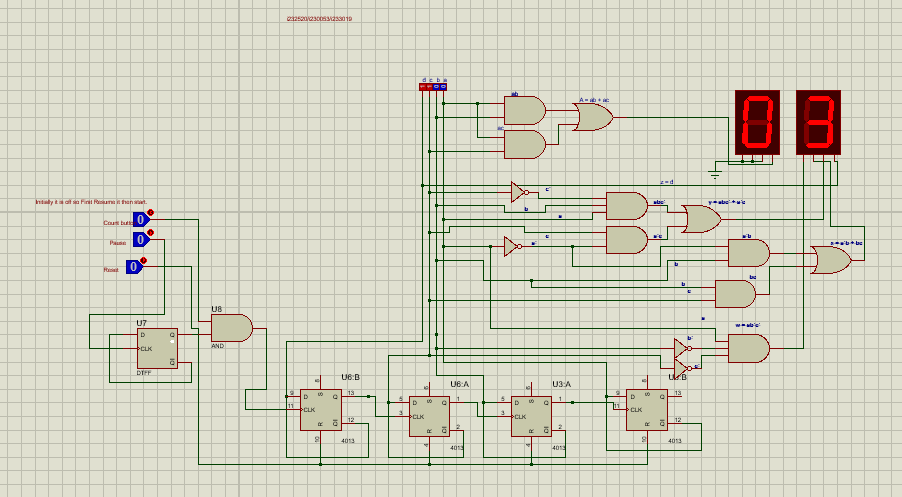
* Proteus

**Hardware:**

* D Flip Flops - 74LS74
* OR Gate IC - 74LS32
* AND Gate IC – 74LS08
* AND-3 Gate IC – 74LS11
* NOT Gate IC – 74LS04
* 7 Segment Display – 7447,7448
* Push Buttons
* Battery (9 Volts)
* Led Lights

**Task4:**

**Circuit Design:**

****