EAST WEST UNIVERSITY

Assignment 2

Course Title: Digital Logic Design

Course Code: CSE345

Section: 04

Submitted to:

Touhid Ahmed

Lecturer

Department of Computer Science & Engineering

Submitted by:

Name: Adri Saha

ID: 2019-1-60-024

Date of submission: 22 May 2021

Topic Name: Synchronous Binary Up Counter

Problem Statement: Design a Synchronous Binary Up Counter which counts 0 to 11

Binary up Counter Design(0 to 11)

As my student id is 2019-1-60-024. Last digit of my Id is 4. So, I have to make binary counter which can count 0 to 11. It can also be called module-12 binary counter as the counter has total 12 states.

Synchronous Counter: The external clock signal is connected to the clock input of every individual flip-flop within the counter in a synchronous counter, causing all of the flip-flops to be clocked together at the same time (in parallel), resulting in a fixed time relationship. Changes in the output occur in "synchronization" with the clock signal, to put it another way.

As a result of this synchronization, all the individual output bits change state at the same time in response to the common clock signal, resulting in no ripple effect and hence no propagation delay.

Module-12 binary counter design: Maximum count of zero 0= $(0000)_2$ to 11= $(1011)_2$ and back to 0 again. As 11 has 4 binary digits, we need 4 bit Up synchronized counter here. Then the 4-Bit counter advances upward in sequence (0,1,2,3,4,5,6,7,8,9,10,11).

Here I used toggle or T flip-flop out of 3 flip flops to design this counter circuit.

First, we need to draw the state diagram:

State diagram: Here total (0-11) total 12 state is present.

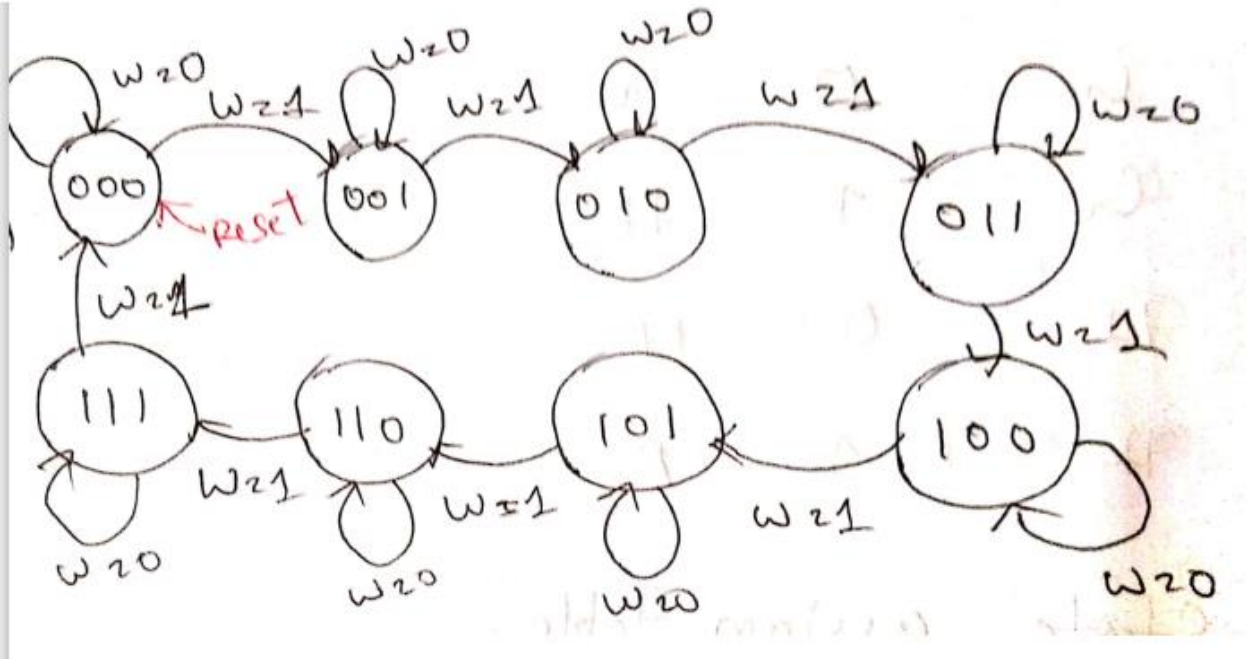

Figure 1: State Diagram of Module-12 binary counter

Assume w is the enable input. When w=0 it holds on the state & when w=1 it simply switches to the next state and count value.

Here maximum number of bits is 4. So, state variables are 4. So, we need 4 flip flops as well.

Now, we take A, B, C, D, …., L to represent the binary states. We can write, for 0 to 11. Binary states are:

A=0000, B=0000, C=0000, D=0000, E=0000, F=0000, G=0000, H=0000, I=0000, J=0000, K=0000, L=0000.

Present State	Input,w	Next state
A	0	A
A	1	B
B	0	B
B	1	C
C	0	C
C	1	D

D	0	D
D	1	E
E	0	E
E	1	F
F	0	F
F	1	G
G	0	G
G	1	H
H	0	H
H	1	I
I	0	I
I	1	J
J	0	J
J	1	K
K	0	K
K	1	L
L	0	L
L	1	A

Figure 2: State Table

State Assigned table:

Now, state assigned table using T flip flop is as follows:

As we know from T flip flop characteristic table T= present state $\oplus$ next state

$=q \oplus Q$. So, I filled up the T flip flop table using ex-or calculation:

	Present State					Next State				Flip Flop			
	q3	q2	q1	q0	w	Q3	Q2	Q1	Q0	T3	T2	T1	T0
A	0	0	0	0	0	0	0	0	0	0	0	0	0
A	0	0	0	0	1	0	0	0	1	0	0	0	1
B	0	0	0	1	0	0	0	0	1	0	0	0	0
B	0	0	0	1	1	0	0	1	0	0	0	1	1
C	0	0	1	0	0	0	0	1	0	0	0	0	0
C	0	0	1	0	1	0	0	1	1	0	0	0	1
D	0	0	1	1	0	0	0	1	1	0	0	0	0
D	0	0	1	1	1	0	1	0	0	0	1	1	1
E	0	1	0	0	0	0	1	0	0	0	0	0	0

E	0	1	0	0	1	0	1	0	1	0	0	0	1
F	0	1	0	1	0	0	1	0	1	0	0	0	0
F	0	1	0	1	1	0	1	1	0	0	0	1	1
G	0	1	1	0	0	0	1	1	0	0	0	0	0
G	0	1	1	0	1	0	1	1	1	0	0	0	1
H	0	1	1	1	0	0	1	1	1	0	0	0	0
H	0	1	1	1	1	1	0	0	0	1	1	1	1
I	1	0	0	0	0	1	0	0	0	1	0	0	0
I	1	0	0	0	1	1	0	0	1	1	0	0	1
J	1	0	0	1	0	1	0	0	1	1	0	0	0
J	1	0	0	1	1	1	0	1	0	1	0	1	1
K	1	0	1	0	0	1	0	1	0	1	0	0	0
K	1	0	1	0	1	1	0	1	1	1	0	0	1
L	1	0	1	1	0	1	0	1	1	1	0	0	0
L	1	0	1	1	1	0	0	0	0	0	0	1	1

Figure 3: State assigned table

Now, **K-map for T flip flops**:

Here we need 4 k-map for T3, T2, T1 & T0.

K-map for T3: we need to draw 2 (4*4 cell k-map) k-map for T3.

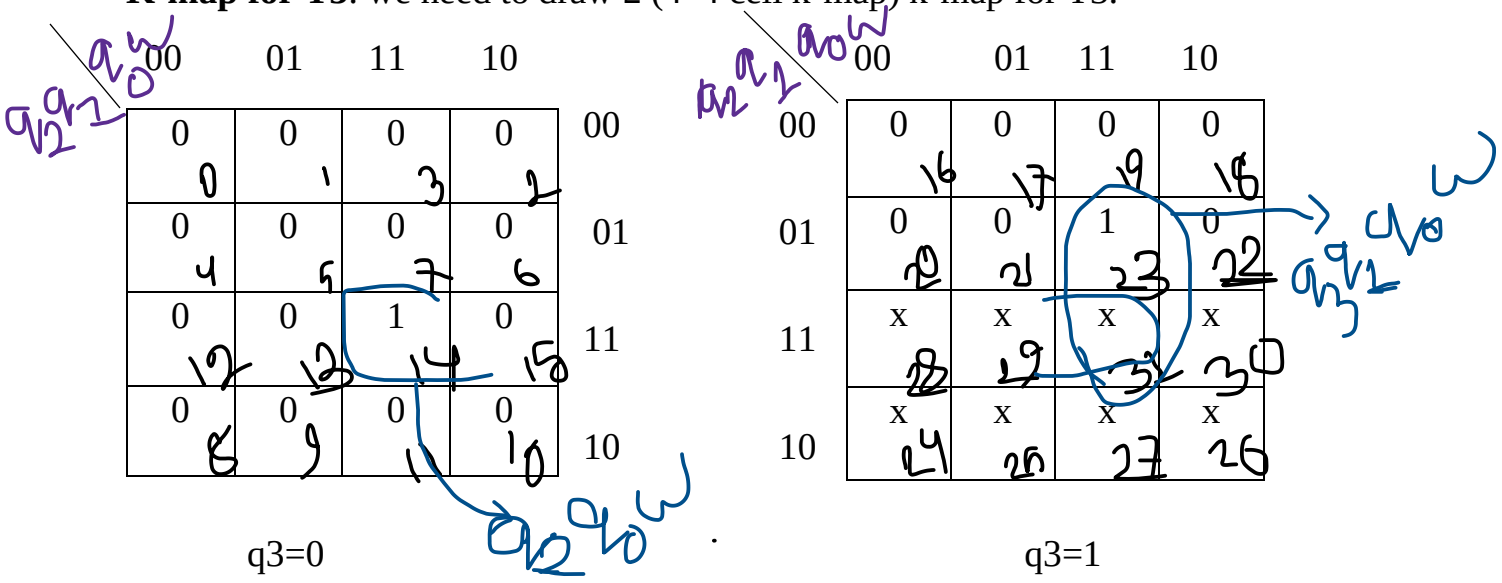

Figure 4: K-map for T3

Here, I draw 2 k-map. One for q3=0 and the other one for q3=1. It is done because of the whole k-map is 24 cells. That is why for simplification we divide into 2 separate k-map where variable q3 is 0 and on another is 1 different. But both k-map are same & for a single variable T3's k-map. When q3=1, the last 2 rows are not needed as I have to design up to $11_{10} = (1011)_2$. So, rest of the cell is don't care. I can take both 0 & 1 on these cells.

So, as the 2 k-map tables are symmetrical here by grouping 14 & 31 cell we get the equation q2.q1.q0.w and by grouping 15 & 30 we get q3.q1.q0.w.

So, T3= q2.q1.q0.w+ q3.q1.q0.w

= q1.q0.w(q2+q3)

K-map for T2: we need to draw 2 (4*4 cell k-map) k-map for T2.

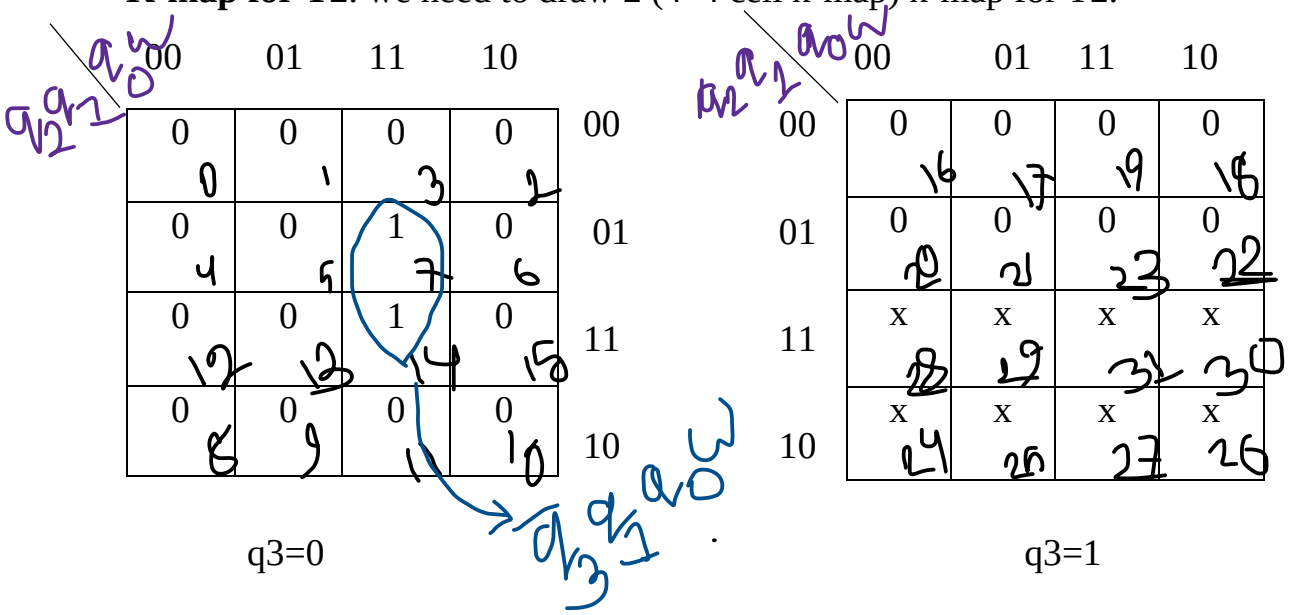

Figure 5: K-map for T2

In this K-map, we find equation only from the q3=0 k-map.

So, T2= $\overline{q3}$.q1.q0.w

K-map for T1: we need to draw 2 (4*4 cell k-map) k-map for T1.

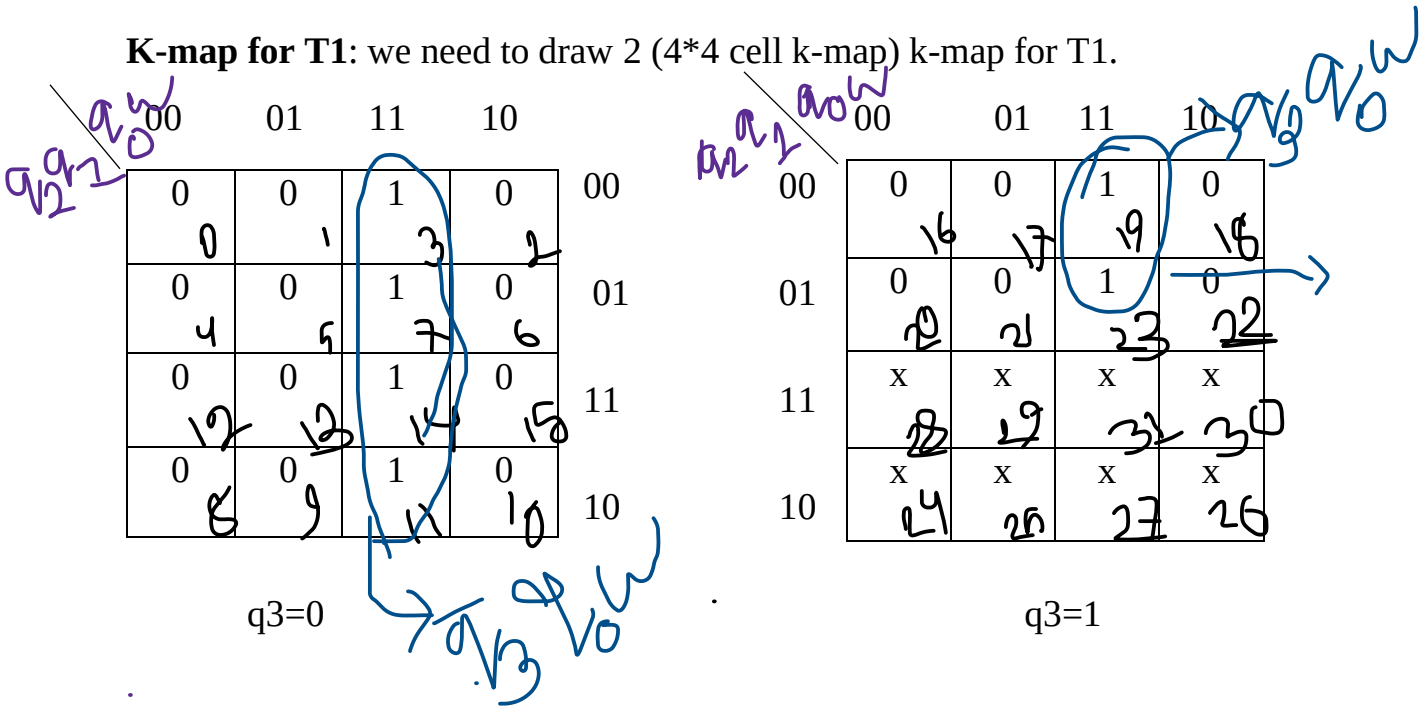

Figure 6: K-map for T1

In this K-map,

So, T1= $\overline{q3}$.q0.w+ q3+q0+w = q0+w ($\overline{q3}$+ q3) = q0+w

K-map for T0: we need to draw 2 (4*4 cell k-map) k-map for each T.

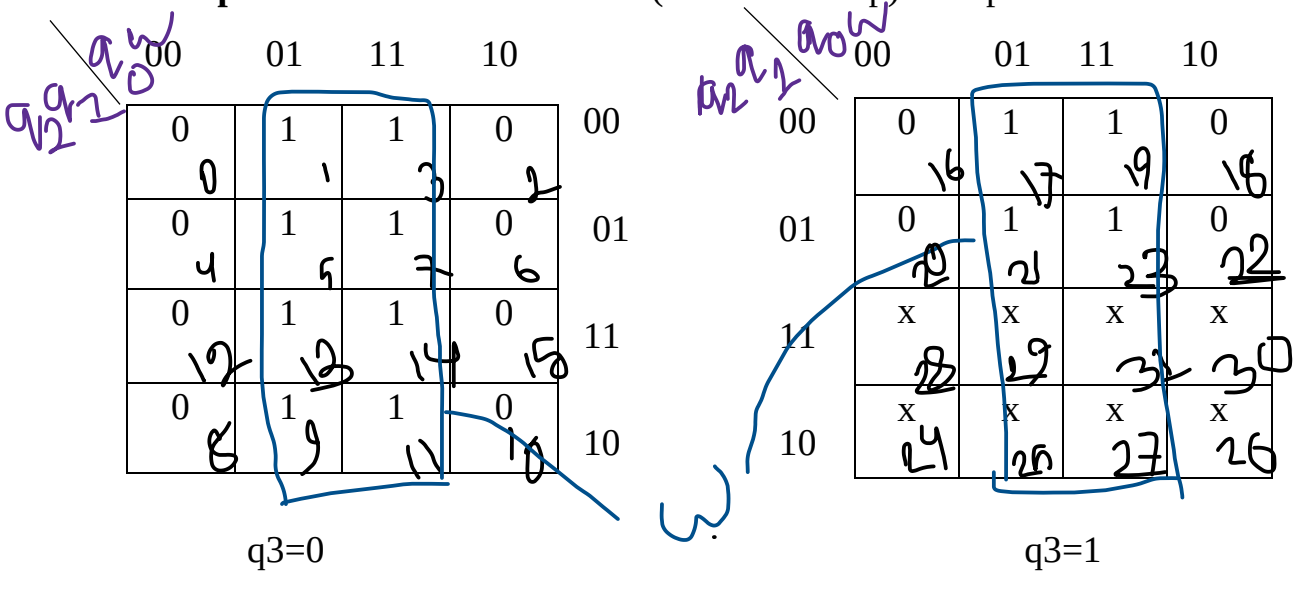

Figure 7: K-map for T0

In this K-map, 1,3,5,7,9,11,17,19,21,23,29,31,25,37 all 16 values are in pair. As they are symmetric.

So, T0= w.

So finally, we ger equations from above k-maps which are:

T3= q1.q0.w(q2+q3),

T2= q3.q1.q0.w,

T1= q0+w,

T0= w.

Now following the equations my design of counter circuit is shown in below:

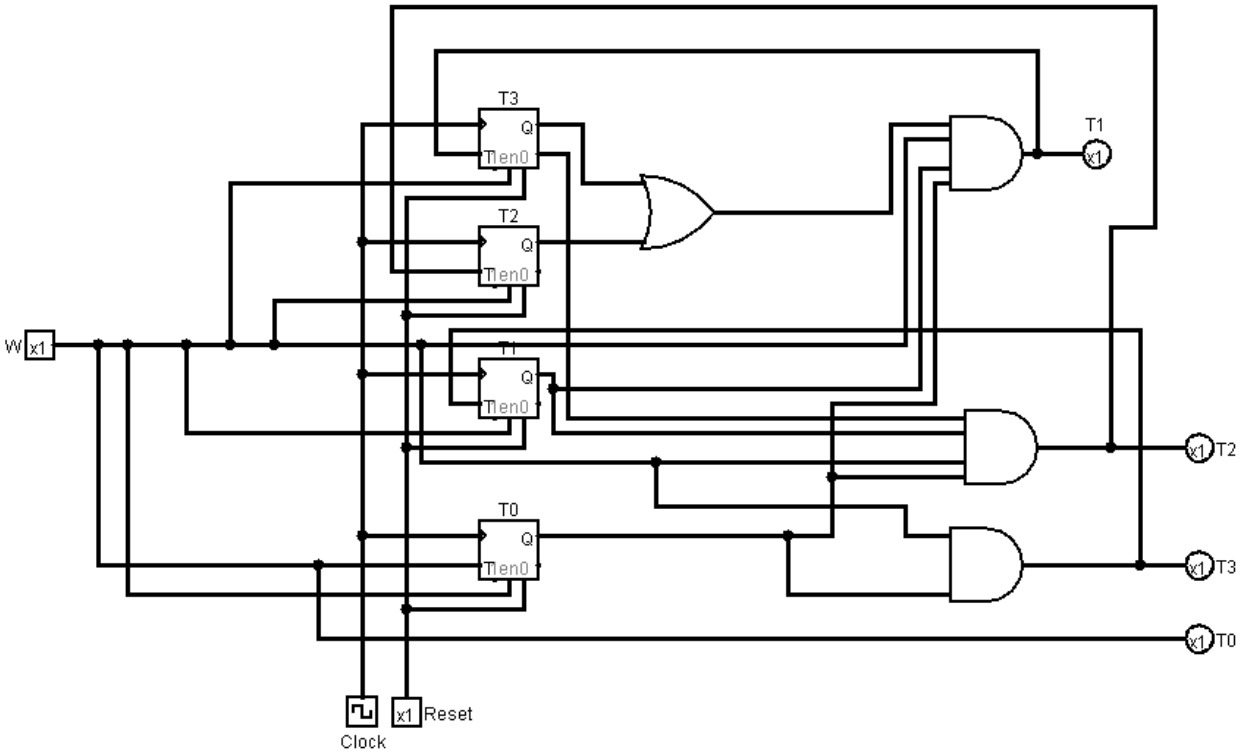

Figure 8: Binary counter circuit design which count up to 11 using T flip-flop

So, this is the desired counter circuit design to count 0 to 11 binary numbers.

After simulating the circuit, we get the simulation result as follows:

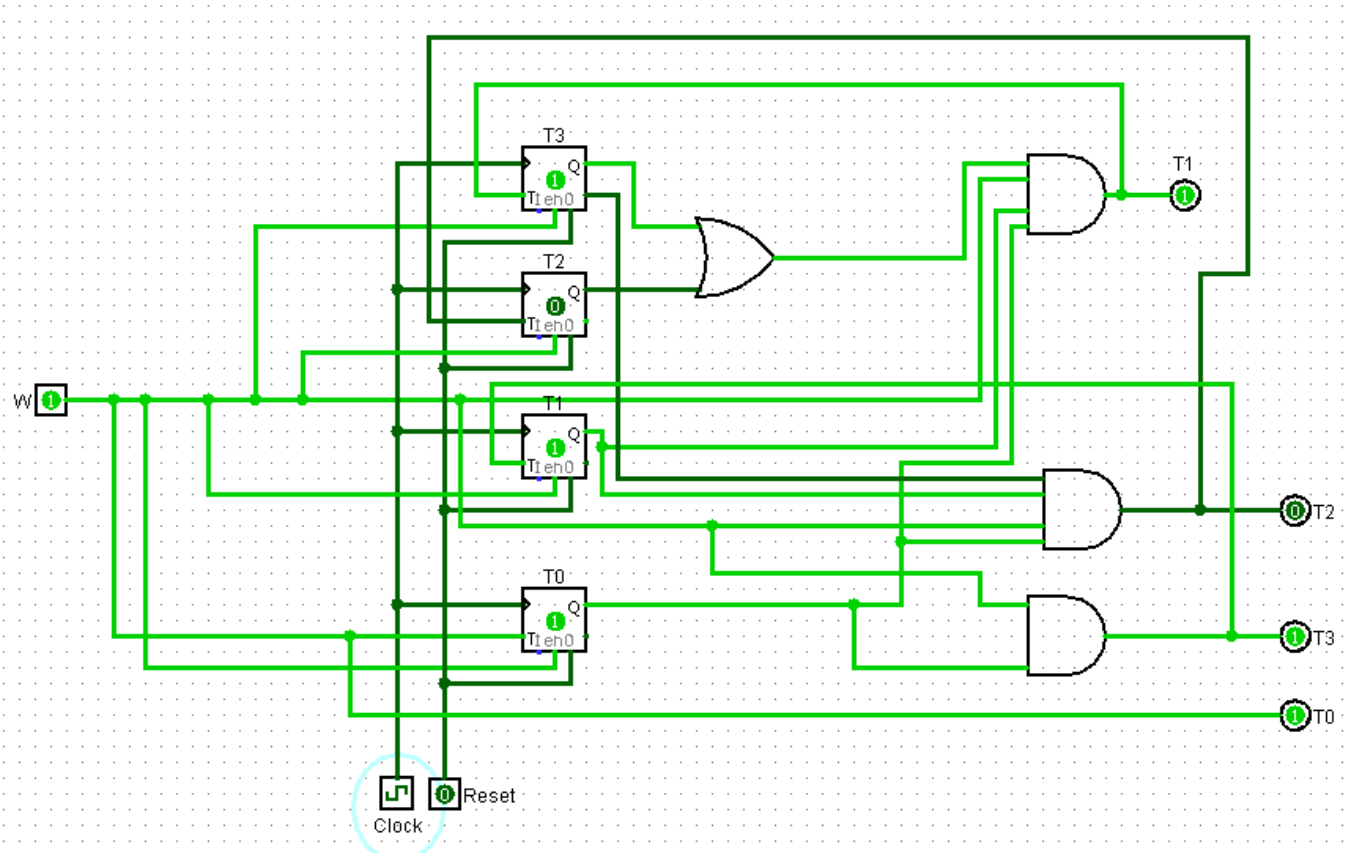

Figure 9: Simulated result of the following binary counter design using T flip-flop

Here, when we change the clock signal T flip flops are change as per as circuit design. And I created some output lines to check the values of T flip flop, which can be verified by the truth table.