DESIGN PROJECT

Ayush Patel

400381354

December 7, 2022

Introduction

Topics 6 and 7 combined with sequential logic design were used to display student number on the 7-segment display. Transition table made for student number is used to makes the k maps. K maps give expressions which help develop logic for the circuit.

The results are simulated in multi sim to confirm the results. These results can further be used to make the physical circuit.

Analytical

Excitation table

Q_n	Q_{n+1}	J	! K	out
0	0	0	X	R or H
0	1	1	X	S or T
1	0	X	0	R or T
1	1	X	1	S or H

This above excitation table was used to make the transition table.

Process 1

Student Number	*Binary*	*counters*
4	0100	00
0	0000	00
0	0000	01
3	0011	00
8	1000	00
1	0001	01
3	0011	01
5	0101	01
4	0100	01

Using 6 variable k maps and transition table

	q1	q2	q3	q4	c5	c6	Q1	Q2	Q3	Q4	C5	C6	J1	!K1	J2	!K2	J3	!K3	J4	!K4	J5	!K5	J6	!K6
4	0	1	0	0	0	0	0	0	0	0	0	0	0	X	X	0	0	X	0	X	0	X	0	X
0	0	0	0	0	0	0	0	0	0	0	0	1	0	X	0	X	0	X	0	X	0	X	1	X
0	0	0	0	0	0	1	0	0	1	1	0	0	0	X	0	X	1	X	1	X	0	X	X	0
3	0	0	1	1	0	0	1	0	0	0	0	0	1	X	0	X	X	0	X	0	0	X	0	X
8	1	0	0	0	0	0	0	0	0	1	0	1	X	0	0	X	0	X	1	X	0	X	1	X
1	0	0	0	1	0	1	0	0	1	1	0	1	0	X	0	X	1	X	X	1	0	X	X	1
3	0	0	1	1	0	1	0	1	0	1	0	1	0	X	1	X	X	0	X	1	0	X	X	1
5	0	1	0	1	0	1	0	1	0	0	0	1	0	X	X	1	0	X	X	0	0	X	X	1
4	0	1	0	0	0	1	0	1	0	0	0	0	0	X	X	1	0	X	0	X	0	X	X	0

State Transition Table – Key with New Coloring Scheme for Easy K-Mapping

	q1	q2	q3	q4	c5	c6	Q1	Q2	Q3	Q4	C5	C6	J1	!K1	J2	!K2	J3	!K3	J4	!K4	J5	!K5	J6	!K6
4	0	1	0	0	0	0	0	0	0	0	0	0	0	X	X	0	0	X	0	X	0	X	0	X
0	0	0	0	0	0	0	0	0	0	0	0	1	0	X	0	X	0	X	0	X	0	X	1	X
0	0	0	0	0	0	1	0	0	1	1	0	0	0	X	0	X	1	X	1	X	0	X	X	0
3	0	0	1	1	0	0	1	0	0	0	0	0	1	X	0	X	X	0	X	0	0	X	0	X
8	1	0	0	0	0	0	0	0	0	1	0	1	X	0	0	X	0	X	1	X	0	X	1	X
1	0	0	0	1	0	1	0	0	1	1	0	1	0	X	0	X	1	X	X	1	0	X	X	1
3	0	0	1	1	0	1	0	1	0	1	0	1	0	X	1	X	X	0	X	1	0	X	X	1
5	0	1	0	1	0	1	0	1	0	0	0	1	0	X	X	1	0	X	X	0	0	X	X	1
4	0	1	0	0	0	1	0	1	0	0	0	0	0	X	X	1	0	X	0	X	0	X	X	0

K-Maps

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00		X		X	00	X	X	X	X
C6=0	01		X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10		X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00				X	00	X	X	X	X

C6=1	01			X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

J1

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	0	X	1	X	00	X	X	X	X
C6=0	01	0	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	X	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	0	0	0	X	00	X	X	X	X
C6=1	01	0	0	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

SOP: c6

! K1

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	X	X	X	X	00	X	X	X	X
C6=0	01	X	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	0	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	X	X	X	X	00	X	X	X	X
C6=1	01	X	X	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

J2

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	0	X	0	X	00	X	X	X	X
C6=0	01	X	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	0	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	0	0	1	X	00	X	X	X	X
C6=1	01	X	X	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

!K2

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	X	X	X	X	00	X	X	X	X
C6=0	01	0	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	X	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	X	X	X	X	00	X	X	X	X
C6=1	01	1	1	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

J3

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	0	X	X	X	00	X	X	X	X
C6=0	01	0	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X

C6=0	10	0	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	1	1	X	X	00	X	X	X	X
C6=1	01	0	0	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

! K3

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	X	X	0	X	00	X	X	X	X
C6=0	01	X	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	X	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	X	X	0	X	00	X	X	X	X
C6=1	01	X	X	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

J4

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	0	X	X	X	00	X	X	X	X
C6=0	01	0	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	1	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	1	X	X	X	00	X	X	X	X
C6=1	01	0	X	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X

C6=1	10	X	X	X	X	10	X	X	X	X

! K4

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	X	X	0	X	00	X	X	X	X
C6=0	01	X	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	X	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	X	1	1	X	00	X	X	X	X
C6=1	01	X	0	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

J5

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	0	X	0	X	00	X	X	X	X
C6=0	01	0	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	0	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	0	0	0	X	00	X	X	X	X
C6=1	01	0	0	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

! K5

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10

C6=0	00	X	X	X	X	00	X	X	X	X
C6=0	01	X	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	X	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	X	X	X	X	00	X	X	X	X
C6=1	01	X	X	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

J6

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	1	X	0	X	00	X	X	X	X
C6=0	01	0	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	1	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=1	00	X	X	X	X	00	X	X	X	X
C6=1	01	X	X	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

! K6

		C5=0	C5=0	C5=0	C5=0		C5=1	C5=1	C5=1	C5=1
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10
C6=0	00	X	X	X	X	00	X	X	X	X
C6=0	01	X	X	X	X	01	X	X	X	X
C6=0	11	X	X	X	X	11	X	X	X	X
C6=0	10	X	X	X	X	10	X	X	X	X
	Q1Q2/Q3Q4	00	01	11	10	Q1Q2/Q3Q4	00	01	11	10

C6=1	00	0	1	1	X	00	X	X	X	X
C6=1	01	0	1	X	X	01	X	X	X	X
C6=1	11	X	X	X	X	11	X	X	X	X
C6=1	10	X	X	X	X	10	X	X	X	X

It is observed that that random bits were taken when the digits were not repeating. This would ultimately lead to a more complex circuit without even trying the consider the POS and SOP expressions.

It can be reduced to 5 variables as digits only repeat at most twice. Process 2 investigates this.

Process 2

Student Number	*Binary*	*counters*
4	0100	0
0	0000	0
0	0000	1
3	0011	0
8	1000	0
1	0001	0
3	0011	1
5	0101	0
4	0100	1

State Transition Table

Using 5 variable k maps and transition table

	q1	q2	q3	q4	c5		Q1	Q2	Q3	Q4	C5	J1	!K1	J2	!K2	J3	!K3	J4	!K4	J5	!K5
4	0	1	0	0	0	0	0	0	0	0	0	0	X	X	0	0	X	0	X	0	X
0	0	0	0	0	0	0	0	0	0	0	1	0	X	0	X	0	X	0	X	1	X
0	0	0	0	0	1	3	0	0	1	1	0	0	X	0	X	1	X	1	X	X	0
3	0	0	1	1	0	8	1	0	0	0	**0**	1	X	0	X	X	0	X	0	0	X
8	1	0	0	0	**0**	1	0	0	0	1	**0**	X	0	0	X	0	X	1	X	0	X
1	0	0	0	1	**0**	3	0	0	1	1	1	0	X	0	X	1	X	X	1	1	X

3	0	0	1	1	1	5	0	1	0	1	0	0	X	1	X	X	0	X	1	X	0
5	0	1	0	1	0	4	0	1	0	0	1	0	X	X	1	0	X	X	0	1	X
4	0	1	0	0	1	4	0	1	0	0	0	0	X	X	1	0	X	0	X	X	0

State Transition Table – Key with New Coloring Scheme for Easy K-Mapping

	q1	q2	q3	q4	c5		Q1	Q2	Q3	Q4	C5	J1	!K1	J2	!K2	J3	!K3	J4	!K4	J5	!K5
4	0	1	0	0	0	0	0	0	0	0	0	0	X	X	0	0	X	0	X	0	X
0	0	0	0	0	0	0	0	0	0	0	1	0	X	0	X	0	X	0	X	1	X
0	0	0	0	0	1	3	0	0	1	1	0	0	X	0	X	1	X	1	X	X	0
3	0	0	1	1	0	8	1	0	0	0	0	1	X	0	X	X	0	X	0	0	X
8	1	0	0	0	0	1	0	0	0	1	0	X	0	0	X	0	X	1	X	0	X
1	0	0	0	1	0	3	0	0	1	1	1	0	X	0	X	1	X	X	1	1	X
3	0	0	1	1	1	5	0	1	0	1	0	0	X	1	X	X	0	X	1	X	0
5	0	1	0	1	0	4	0	1	0	0	1	0	X	X	1	0	X	X	0	1	X
4	0	1	0	0	1	4	0	1	0	0	0	0	X	X	1	0	X	0	X	X	0

K maps

Q4c5\q1q2q3	000	001	011	010	100	101	111	110
00		X	X			X	X	X
01		X	X		X	X	X	X
11	X		X	X	X	X	X	X
10			X		X	X	X	X

J1

q4c5\q1q2q3	000	001	011	010	100	101	111	110
00	0	X	X	0	X	X	X	X
01	0	X	X	0	X	X	X	X

11	X	0	X	X	X	X	X	X
10	0	1	X	0	X	X	X	X

SOP: q3c5! 1 AND NAND: (q3c5!)!! => 2 NAND

POS: q3q5! 1 AND

! K1

q4c5\q1q2q3	**000**	**001**	**011**	**010**	**100**	**101**	**111**	**110**
00	X	X	X	X	0	X	X	X
01	X	X	X	X	X	X	X	X
11	X	X	X	X	X	X	X	X
10	X	X	X	X	X	X	X	X

SOP: ground NAND: => 0 NAND

POS: ground

J2

Q4c5\q1q2q3	**000**	**001**	**011**	**010**	**100**	**101**	**111**	**110**
00	0	X	X	X	0	X	X	X
01	0	X	X	X	X	X	X	X
11	X	1	X	X	X	X	X	X
10	0	0	X	X	X	X	X	X

SOP: q4 c5 1 AND NAND: (q4c5)!! => 2 NAND

POS: q3 c5 1 AND

! K2

Q4c5\q1q2q3	**000**	**001**	**011**	**010**	**100**	**101**	**111**	**110**

00	X	X	X	0	X	X	X	X
01	X	X	X	1	X	X	X	X
11	X	X	X	X	X	X	X	X
10	X	X	X	1	X	X	X	X

SOP: q4 + c5 1 OR NAND: (q4! q5!)! => 1 NAND

POS: q4 + q5 1 OR

J3

Q4c5\q1q2q3	**000**	**001**	**011**	**010**	**100**	**101**	**111**	**110**
00	0	X	X	0	0	X	X	X
01	1	X	X	0	X	X	X	X
11	X	X	X	X	X	X	X	X
10	1	X	X	0	X	X	X	X

SOP: q2! c5 + q2! q4 => q2! (c5 + q4) 1 OR and 1 AND

NAND: ((q2! c5)! (q2! c4)!)! => 3 NAND

POS: q2! (q4 + q5) 1 OR and 1 AND

! K3

Q4c5\q1q2q3	**000**	**001**	**011**	**010**	**100**	**101**	**111**	**110**
00	X	X	X	X	X	X	X	X
01	X	X	X	X	X	X	X	X
11	X	0	X	X	X	X	X	X
10	X	0	X	X	X	X	X	X

SOP: ground

NAND: => 0 NAND

POS: q4!

J4

Q4c5\q1q2q3	000	001	011	010	100	101	111	110
00	0	X	X	0	1	X	X	X
01	1	X	X	0	X	X	X	X
11	X	X	X	X	X	X	X	X
10	X	X	X	X	X	X	X	X

SOP: q1 + (q2! c5) => 1 OR and 1 AND

NAND: ((q1!) (q2! c5)!!)! => 3 NAND

POS: (q2!) (q1 + c5) => 1 OR and 1 AND

! K4

Q4c5\q1q2q3	000	001	011	010	100	101	111	110
00	X	X	X	X	X	X	X	X
01	X	X	X	X	X	X	X	X
11	X	1	X	X	X	X	X	X
10	1	0	X	0	X	X	X	X

SOP: c5 + q2! q3! => 1 OR and 1 AND

NAND: ((c5)! (q2! q3!)!)! => 2 NANDS

POS: (q2!) (q3! + c5) => 1 OR and 1 AND

J5

Q4c5\q1q2q3	000	001	011	010	100	101	111	110
00	1	X	X	0	0	X	X	X
01	X	X	X	X	X	X	X	X

11	X	X	X	X	X	X	X	X
10	1	0	X	1	X	X	X	X

SOP: q3! q4 + q1! q2! q3!=>q3! (q4 + q1! q2!) 2 AND and 1 OR

NAND: (q3! ((q4! (q1! q2!)!))!)!! => 4 NAND

POS: (q3!) (q1!) (q2! + q4) => 2 AND and 1 OR

! K5

Q4c5\q1q2q3	**000**	**001**	**011**	**010**	**100**	**101**	**111**	**110**
00	X	X	X	X	X	X	X	X
01	0	X	X	0	X	X	X	X
11	X	0	X	X	X	X	X	X
10	X	X	X	X	X	X	X	X

SOP: ground => 0 NAND

POS: ground

Table for all the types of implementations

PIN	**SOP**	**NAND for SOP**	**POS**
J1	q3c5!	(q3c5!)!!	q3c5!
! K1	GND	GND	GND
J2	q4 q5	(q4c5)!!	q3 q5
! K2	q4 + q5	(q4! q5!)!	q4 + q5
J3	q2! c5 + q2! q4	((q2! c5)! (q2! c4)!)!	q2! (q4 + q5)
! K3	GND	GND	q4!
J4	q1 + (q2! c5)	((q1!) (q2! c5)!!)!	(q2!) (q1 + c5)
! K4	c5 + q2! q3!	((c5)! (q2! q3!)!)!	(q2!) (q3! + c5)

J5	q3! q4 + q1! q2! q3!	(q3!((q4!(q1!q2!)!))!)!!	(q3!) (q1!) (q2! + q4)
! K5	GND	GND	GND

5 OR and 7 AND => 4 chips

5 FLIP FLOPS => 2 chips

total 6 chips

Process 3

For process 3, we can try the NAND implementation of SOP.

17 NANDs => 5 chips

5 FLIP FLOPS => 2 chips

total 7 chips

Conclusion

It is more space efficient to do **AND/OR** implementation **of SOP for 5 variable** k maps since it uses **6 chips** instead of 7. In general NAND implementation is preferred but in this case, what process 2 gives looks more efficient which saves space on the bread board.

Therefore, multi sim implementation has been performed using **process 2.**

Multi sim

Timing diagram provided by the XLA1 logic analyzer helps understand how the logic is changing. All flip flops are connected to a synchronized clock. The circuit the voltage limits and maintained so the circuit does not break down.

Color scheme:

Red: FF1

Blue: FF2

Green: FF3

Pink: FF4

Golden brown: FF5

Black: clock

Also Red: 7 seg display wires

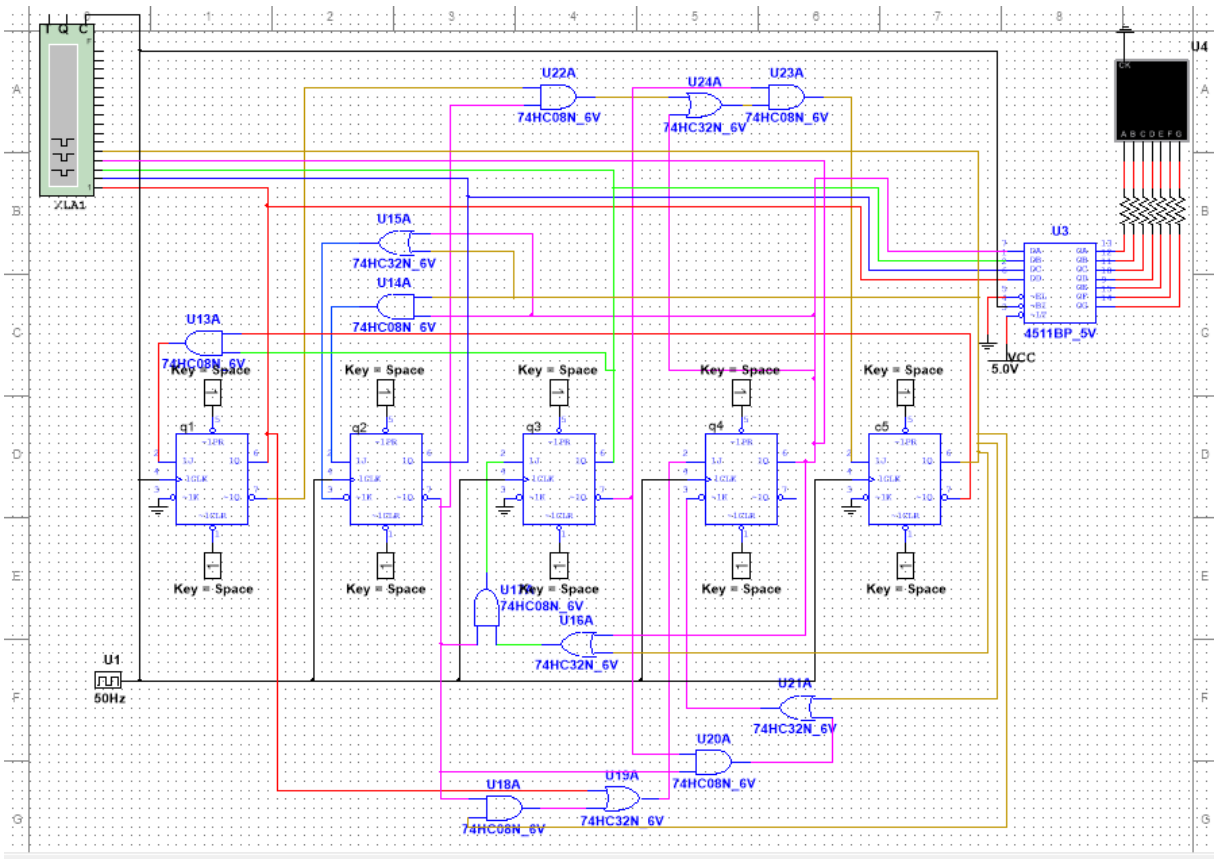

Fig. Multisim

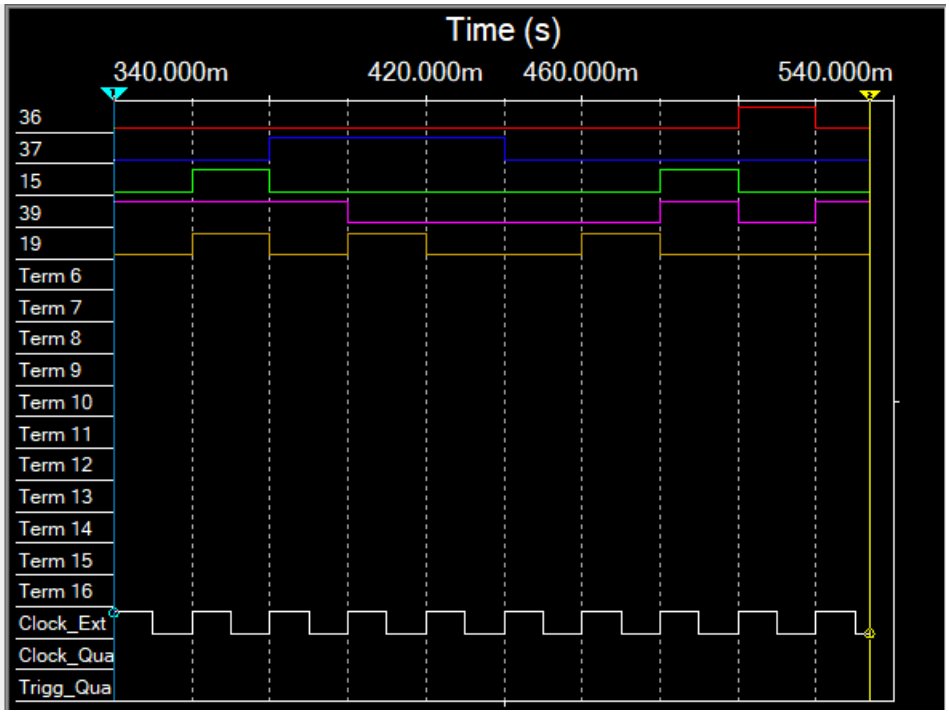

Fig. Timing diagram from multi sim circuit