

# 컴퓨터 공학 기초 실험2 보고서

실험제목: Traffic Light Controller with/without  
Left Turn Signals

실험일자: 2023년 10월 16일 (월)

제출일자: 2023년 10월 25일 (수)

학 과: 컴퓨터공학과

담당교수: 이준환 교수님

실습분반: 월요일 0, 1, 2

학 번: 20222020264

성 명: 최봉규

## 1. 제목 및 목적

### A. 제목

Traffic Light Controller with/without Left Turn Signals

### B. 목적

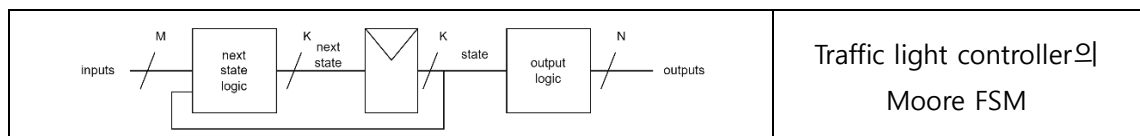
이번 실습에서는 신호등을 제어하는 Logic을 구현한다. Sequential logic을 Verilog로 짜는 방법을 학습하고 이를 구현해본다. Verilog의 코드 구현 방식인 Behavior과 struct 방식을 이해하고 이를 이용해 두 가지의 traffic light controller를 구현한다.

## 2. 원리(배경지식)

신호등  $L_A$ 는 'Academic Ave.'의 차량 통행을 제어하는 신호등이고,  $L_B$ 는 'Bravado Blvd.'의 차량 통행을 제어하는 신호등이다. 일반적인 신호등의 경우는 시간에 따라 변하지만, 이번 실습을 통해 구현하려는 신호등은 거리에 차량이 있을 때 신호등이 초록색이 되고, 없을 때는 빨간색이 된다. 차량이 있음을 감지하기 위하여 'Academic Ave.'에 traffic sensor인  $T_A$ 를, 'Bravado Blvd.'에 traffic sensor인  $T_B$ 를 설치한다.

신호등의 구현을 위해 다음 규칙을 만족해야 한다.

- Traffic light는 교통이 없을 때 초록색에서 노란색을 거쳐 빨간색으로 변한다.
- 만약 traffic light  $L_A$ 가 초록색이거나 노란색이면, traffic light  $L_B$ 는 빨간색이다. 반대의 경우에도 마찬가지이다.



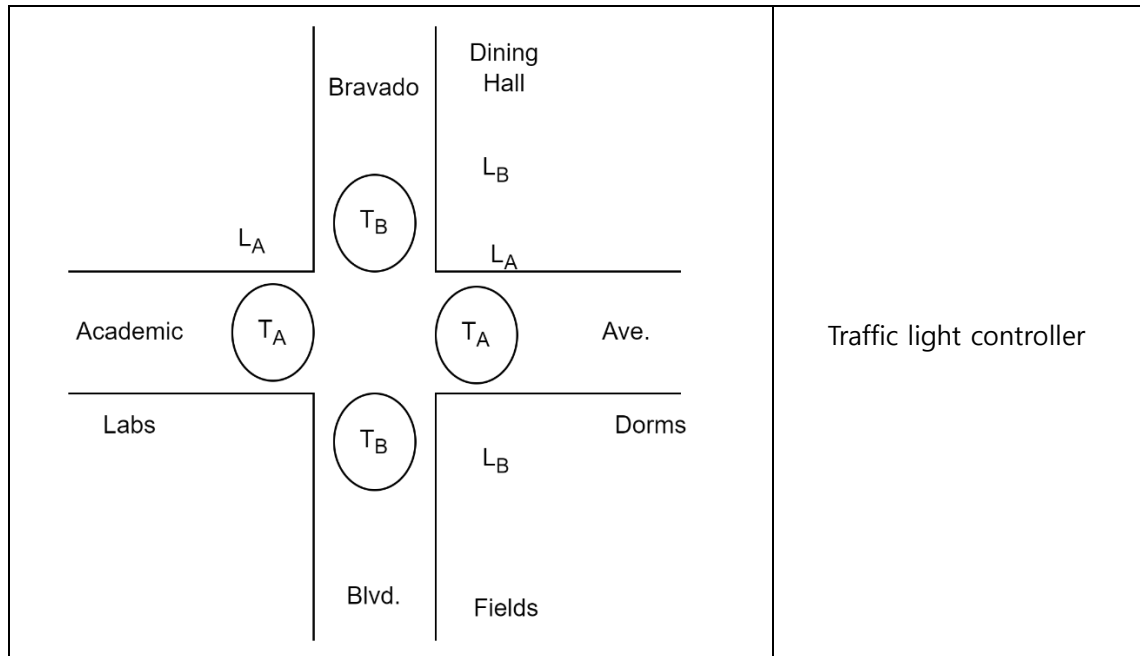
신호등을 제어하는 logic을 구현한다. 앞서 실습한 traffic light controller에 left turn signal을 추가하여 구현한다. 신호등  $L_A$ 는 'Academic Ave.'의 차량 통행을 제어하는 신호등이다.  $L_B$ 는 'Bravado Blvd.'의 차량 통행을 제어하는 신호등이다. 시간에 따라 변하는 신호등이 아닌 거리에 차량이 있을 때 신호등이 초록색이 되고, 없을 때는 빨간색이 된다. 차량이 있음을 감지하기 위하여 'Academic Ave.'에 traffic sensor인  $T_A$ ,  $T_B$ 는 직진에 대한 차량을 감시한다. Traffic sensor인  $T_{AL}$ ,  $T_{BL}$ 은 좌회전에 대한 차량을 감시한다.

신호등의 구현은 다음을 만족한다.

- Traffic light는 교통이 없을 때 초록색에서 노란색을 거쳐 좌회전으로 변한다.
- Traffic light는 교통이 없을 때 좌회전에서 노란색을 거쳐 빨간색으로 변한다.
- Traffic light는 비록 좌회전하는 교통이 없더라도 초록색에서 좌회전으로 우선 변해

야 한다.

- 만약 traffic light  $L_A$ 가 초록색, 노란색, 좌회전일 동안에  $L_B$ 는 빨간색이어야 한다. 반대 경우도 마찬가지이다.

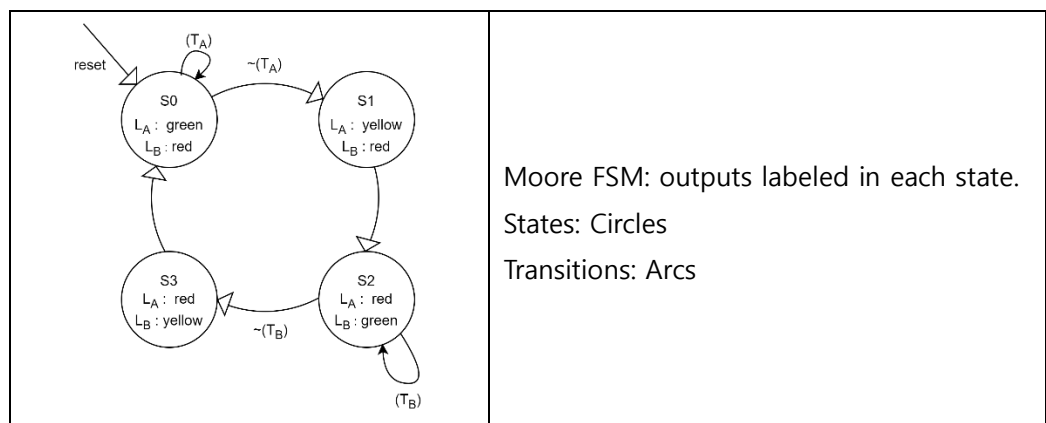


### 3. 설계 세부사항

#### 1) Traffic Light Controller

Define states	S0, S1, S2, S3
Define inputs	$T_A$ , $T_B$
Define outputs	$L_{A1}$ , $L_{A0}$ , $L_{B1}$ , $L_{B0}$

Draw the diagram



Encoding states

Current state	$Q_1$	$Q_0$
S0	0	0
S1	0	1
S2	1	1
S3	1	0

FSM Encoded State Transition Table

Current state		Inputs		Next state	
$Q_1$	$Q_0$	$T_A$	$T_B$	$D_1$	$D_0$
0	0	0	X	0	1
0	0	1	X	0	0
0	1	X	X	1	0
1	0	X	0	1	1
1	0	X	1	1	0
1	1	X	X	0	0

Boolean equation	$D_1 = Q_1 \oplus Q_0$ $D_0 = !Q_1!Q_0!T_A + Q_1!Q_0!T_B$
------------------	---

FSM output table

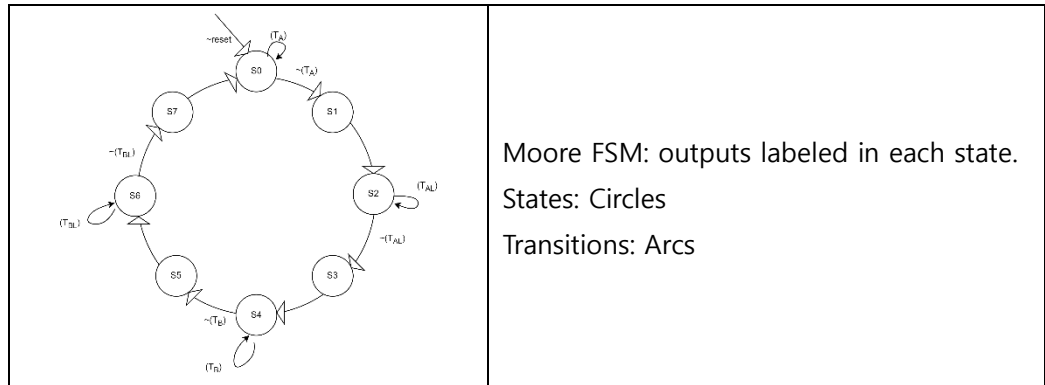
Current state		Outputs			
$Q_1$	$Q_0$	$L_{A1}$	$L_{A0}$	$L_{B1}$	$L_{B0}$
0	0	0	0	1	0
0	1	0	1	1	0
1	0	1	0	0	0
1	1	1	0	0	1

Boolean equation	$L_{A1} = Q_1$ $L_{A0} = Q_1Q_0$ $L_{B1} = \sim Q_1$ $L_{B0} = Q_1Q_0$
------------------	--

## 2) Traffic Light Controller with Left Turn signals

Define states	S0, S1, S2, S3, S4, S5, S6, S7
Define inputs	$T_A$ , $T_B$ , $T_{AL}$ , $T_{BL}$
Define outputs	$L_{A1}$ , $L_{A0}$ , $L_{B1}$ , $L_{B0}$

Draw the diagram



Encoding states

Current state	$Q_2$	$Q_1$	$Q_0$
S0	0	0	0
S1	0	0	1
S2	0	1	0
S3	0	1	1
S4	1	0	0
S5	1	0	1
S6	1	1	0
S7	1	1	1

FSM Encoded State Transition Table

Current state			Input				Next state		
$Q_2$	$Q_1$	$Q_0$	$T_A$	$T_{AL}$	$T_B$	$T_{BL}$	$D_2$	$D_1$	$D_0$
0	0	0	0	X	X	X	0	0	1
0	0	0	1	X	X	X	0	0	0
0	0	1	X	X	X	X	0	1	0
0	1	0	X	0	X	X	0	1	1
0	1	0	X	1	X	X	0	1	0
0	1	1	X	X	X	X	1	0	0
1	0	0	X	X	0	X	1	0	1
1	0	0	X	X	1	X	1	0	0
1	1	0	X	X	X	0	1	1	1
1	1	0	X	X	X	1	1	1	0
1	1	1	X	X	X	X	0	0	0

QM

D <sub>2</sub>										
Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Column 8	Column 9	Column 10	Column 11
1	100000	1	100000	1	1000-0	1	100-0	1	100----	1
2	0110000	1	1000-00	1	100-00	1	1000---	1	10---0-	1
2	1000001	1	100-000	1	10000--	1	10-0-0-	1	1-0-0-	
2	1000100	1	10000-0	1	10-000-	1	1-00-0-	1	10-0---	
2	1001000	1	10-0000	1	1-0000-	1	100-0--	1	1-00---	
2	1000010	1	1-00000	1	100--00	1	10--00-	1	10--0-	
2	1010000	2	011000-	1	1000-0	1	1-0-00-	1	1-0-0-	
2	1100000	2	01100-0	1	10-0-00	1	10-00--	1	10---0	
3	0110001	2	0110-00	1	1-00-00	1	1-000--	1	1-0-0-	
3	0110010	2	011-000	1	100-0-0	1	100--0	2	011---	
3	0110100	2	1000-01	1	10--000	1	10--00	2	10---1	
3	0111000	2	100-001	1	1-0-000	1	1-0--00	2	1-0--1	
3	1000101	2	10000-1	1	10-00-0	1	10-0-0	2	10-1--	
3	1001001	2	10-0001	1	1-000-0	1	1-00-0	2	1-0-1--	
3	1001100	2	1-00001	2	01100--	1	10--0-0	2	1-0-1---	
3	1000011	2	100010-	2	0110-0-	1	1-0-0-0	2	1-01---	
3	1000110	2	100-100	2	011-00-	2	0110---	2	10---1-	
3	1001010	2	10001-0	2	0110--0	2	011-0--	2	1-0-1-	
3	1010001	2	10-0100	2	011-0-0	2	011--0-	2	101---	
3	1010010	2	1-00100	2	011--00	2	011--0	2	110---	
3	1010100	2	100100-	2	100--01	2	100--1			
3	1011000	2	1001-00	2	1000-1	2	10--01			
3	1100001	2	10010-0	2	10-0-01	2	1-0-01			
3	1100010	2	10-1000	2	1-00-01	2	10-0-1			
3	1100100	2	1-01000	2	100-0-1	2	1-00-1			
3	1101000	2	100001-	2	10--001	2	10--0-1			
4	0110011	2	1000-10	2	1-0-001	2	1-0-0-1			
4	0110101	2	100-010	2	10-00-1	2	100-1--			
4	0110110	2	10-0010	2	1-000-1	2	10--10-			
4	0111001	2	1-00010	2	100-10-	2	1-0-10-			
4	0111010	2	101000-	2	10001--	2	10-01--			
4	0111100	2	10100-0	2	10-010-	2	1-001--			
4	1001101	2	1010-00	2	1-0010-	2	10--1-0			
4	1000111	2	101-000	2	100-1-0	2	1-0-1-0			
4	1001011	2	110000-	2	10--100	2	1001---			
4	1001110	2	11000-0	2	1-0-100	2	10-1-0-			
4	1010011	2	1100-00	2	10-01-0	2	1-01-0-			
4	1010101	2	110-000	2	1-001-0	2	10-10--			
4	1010110	3	01100-1	2	1001-0-	2	1-010--			
4	1011001	3	0110-01	2	10010--	2	10-1-0			
4	1011010	3	011-001	2	10-100-	2	1-01-0			
4	1011100	3	011001-	2	1-0100-	2	100--1-			
4	1100011	3	0110-10	2	1001-0	2	10-0-1-			
4	1100101	3	011-010	2	10-1-00	2	1-00-1-			
4	1100110	3	011010-	2	1-01-00	2	10--01-			
4	1101001	3	01101-0	2	10-10-0	2	1-0-01-			
4	1101010	3	011-100	2	1-010-0	2	10--10			
4	1101100	3	011100-	2	1000-1-	2	1-0-10			
5	0110111	3	01110-0	2	100-01-	2	1010---			
5	0111011	3	0111-00	2	10-001-	2	101-0-			
5	0111101	3	100-101	2	1-0001-	2	101--0-			
5	0111110	3	10001-1	2	100--10	2	101--0			
5	1001111	3	10-0101	2	10-0-10	2	1100---			
5	1010111	3	1-00101	2	1-00-10	2	110-0-			
5	1011011	3	1001-01	2	10--010	2	110--0-			
5	1011101	3	10010-1	2	1-0-010	2	110--0			
5	1011110	3	10-1001	2	10100--	3	011---1			
5	1100111	3	1-01001	2	1010-0-	3	011--1-			
5	1101011	3	100110-	2	101-00-	3	011-1--			
5	1101101	3	10011-0	2	1010--0	3	0111---			
5	1101110	3	10-1100	2	101-0-0	3	10--1-1			
6	0111111	3	1-01100	2	101--00	3	1-0-1-1			
6	1011111	3	1000-11	2	11000--	3	10-1-1-			
6	1101111	3	100-011	2	1100-0-	3	1-01-1			
		3	10-0011	2	110-00-	3	10-11--			
		3	1-00011	2	1100--0	3	1-011--			
		3	100011-	2	110-0-0	3	10---11			
		3	100-110	2	110--00	3	1-0--11			
		3	10-0110	3	0110--1	3	10--11-			
		3	1-00110	3	011-0-1	3	1-0-11-			
		3	100101-	3	011--01	3	10-1-1-			
		3	1001-10	3	0110-1-	3	1-01-1-			
		3	10-1010	3	011-01-	3	101---1			
		3	1-01010	3	011--10	3	101--1-			
		3	10100-1	3	01101--	3	101-1--			
		3	1010-01	3	011-10-	3	1011---			
		3	101-001	3	011-1-0	3	110---1			
		3	101001-	3	01110--	3	110--1-			
		3	1010-10	3	0111-0-	3	110-1--			
		3	101-010	3	0111--0	3	1101---			
		3	101010-	3	100-1-1					
		3	10101-0	3	10--101					
		3	101-100	3	1-0-101					
		3	101100-	3	10-01-1					
		3	10110-0	3	1-001-1					
		3	1011-00	3	1001--1					
		3	11000-1	3	10-1-01					
		3	1100-01	3	1-01-01					
		3	110-001	3	10-10-1					
		3	110001-	3	1-010-1					
		3	1100-10	3	10011--					
		3	110-010	3	10-110-					
		3	110010-	3	1-0110-					
		3	11001-0	3	10-11-0					

	3	110-100	3	1-011-0
	3	110100-	3	100--11
	3	11010-0	3	10-0-11
	3	1101-00	3	1-00-11
	4	0110-11	3	10--011
	4	011-011	3	1-0-011
	4	01101-1	3	100-11-
	4	011-101	3	10-011-
	4	011011-	3	1-0011-
	4	011-110	3	10--110
	4	01110-1	3	1-0-110
	4	0111-01	3	1001-1-
	4	011101-	3	10-101-
	4	0111-10	3	1-0101-
	4	011110-	3	10-1-10
	4	01111-0	3	1-01-10
	4	10011-1	3	1010--1
	4	10-1101	3	101-0-1
	4	1-01101	3	101--01
	4	100-111	3	1010-1-
	4	10-0111	3	101-01-
	4	1-00111	3	101--10
	4	1001-11	3	10101--
	4	10-1011	3	101-10-
	4	1-01011	3	101-1-0
	4	100111-	3	10110--
	4	10-1110	3	1011-0-
	4	1-01110	3	1011--0
	4	1010-11	3	1100--1
	4	101-011	3	110-0-1
	4	10101-1	3	110--01
	4	101-101	3	1100-1-
	4	101011-	3	110-01-
	4	101-110	3	110--10
	4	10110-1	3	11001--
	4	1011-01	3	110-10-
	4	101101-	3	110-1-0
	4	1011-10	3	11010--
	4	101110-	3	1101-0-
	4	10111-0	3	1101--0
	4	1100-11	4	011--11
	4	110-011	4	011-1-1
	4	11001-1	4	011-11-
	4	110-101	4	0111--1
	4	110011-	4	0111-1-
	4	110-110	4	01111--
	4	11010-1	4	10-11-1
	4	1101-01	4	1-011-1
	4	110101-	4	10--111
	4	1101-10	4	1-0-111
	4	110110-	4	10-1-11
	4	11011-0	4	1-01-11
	5	011-111	4	10-111-
	5	0111-11	4	1-0111-
	5	01111-1	4	101--11
	5	011111-	4	101-1-1
	5	10-1111	4	101-11-
	5	1-01111	4	1011--1
	5	101-111	4	1011-1-
	5	1011-11	4	10111--
	5	10111-1	4	110--11
	5	101111-	4	110-1-1
	5	110-111	4	110-11-
	5	1101-11	4	1101--1
	5	11011-1	4	1101-1-
	5	110111-	4	11011--
Prime Implicants			Essential Prime Implicants	
011----			011----	
10-----			10-----	
1-0----			1-0----	

D<sub>1</sub>

Column 1		Column 2		Column 3		Column 4		Column 5		Column 6	
1	0010000	1	001000-	1	00100--	1	0010---	1	001----	1	-01----
1	0100000	1	00100-0	1	0010-0-	1	001-0--	1	-010---	1	-10----
2	0010001	1	0010-00	1	001-00-	1	-0100--	1	-01-0--		
2	0010010	1	001-000	1	-01000-	1	001--0-	1	-01--0-		
2	0010100	1	-010000	1	0010--0	1	-010-0-	1	-01---0		
2	0011000	1	010000-	1	001-0-0	1	-01-00-	1	010----		
2	0100001	1	01000-0	1	-0100-0	1	001---0	1	-100---		
2	0100010	1	0100-00	1	001--00	1	-010--0	1	-10-0--		
2	0100100	1	010-000	1	-010-00	1	-01-0-0	1	-10--0-		
2	0101000	1	-100000	1	-01-000	1	-01--00	1	-10---0		
2	1010000	2	00100-1	1	01000--	1	0100---	2	-01---1		
2	1100000	2	0010-01	1	0100-0-	1	010-0--	2	-01--1-		
3	0010011	2	001-001	1	010-00-	1	-1000--	2	-01-1--		
3	0010101	2	-010001	1	-10000-	1	010--0-	2	-011---		
3	0010110	2	001001-	1	0100--0	1	-100-0-	2	-10---1		
3	0011001	2	0010-10	1	010-0-0	1	-10-00-	2	-10--1-		
3	0011010	2	001-010	1	-1000-0	1	010---0	2	-101---		
3	0100011	2	001010-	1	-100-00	1	-10-0-0	2	101----		
3	0100101	2	00101-0	1	-10-000	1	-10--00	2	110----		
3	0100110	2	001-100	2	0010--1	2	001---1				
3	0101001	2	-010100	2	001-0-1	2	-010--1				
3	0101010	2	001100-	2	-0100-1	2	-01-0-1				
3	0101100	2	00110-0	2	001--01	2	-01--01				
3	1010001	2	0011-00	2	-010-01	2	001--1-				
3	1010010	2	-011000	2	-01-001	2	-010-1-				
3	1010100	2	01000-1	2	0010-1-	2	-01-01-				
3	1011000	2	0100-01	2	001-01-	2	-01--10				
3	1100001	2	010-001	2	-01001-	2	001-1--				
3	1100010	2	-100001	2	001--10	2	-0101--				
3	1100100	2	010001-	2	-010-10	2	-01-10-				
3	1101000	2	0100-10	2	-01-010	2	-01-1-0				
4	0010111	2	010-010	2	00101--	2	0011---				
4	0011011	2	-100010	2	001-10-	2	-0110--				
4	0011101	2	010010-	2	-01010-	2	-011-0-				
4	0011110	2	01001-0	2	001-1-0	2	-011--0				
4	0100111	2	010-100	2	-0101-0	2	010---1				
4	0101011	2	-100100	2	-01-100	2	-100--1				
4	0101101	2	010100-	2	00110--	2	-10-0-1				
4	0101110	2	01010-0	2	0011-0-	2	-10--01				
4	1010011	2	0101-00	2	-01100-	2	010--1-				
4	1010101	2	-101000	2	0011--0	2	-100-1-				
4	1010110	2	101000-	2	-0110-0	2	-10-01-				
4	1011001	2	10100-0	2	-011-00	2	-10--10				
4	1011010	2	1010-00	2	0100--1	2	010-1--				
4	1011100	2	101-000	2	010-0-1	2	-1001--				
4	1100011	2	110000-	2	-1000-1	2	-10-10-				
4	1100101	2	11000-0	2	010--01	2	-10-1-0				
4	1100110	2	1100-00	2	-100-01	2	0101---				
4	1101001	2	110-000	2	-10-001	2	-1010--				
4	1101010	3	0010-11	2	0100-1-	2	-101-0-				
4	1101100	3	001-011	2	010-01-	2	-101--0				
5	0011111	3	-010011	2	-10001-	2	1010---				
5	0101111	3	00101-1	2	010--10	2	101-0--				
5	1010111	3	001-101	2	-100-10	2	101--0-				
5	1011011	3	-010101	2	-10-010	2	101---0				
5	1011101	3	001011-	2	01001--	2	1100---				
5	1011110	3	001-110	2	010-10-	2	110-0--				
5	1100111	3	-010110	2	-10010-	2	110--0-				
5	1101011	3	00110-1	2	010-1-0	2	110---0				
5	1101101	3	0011-01	2	-1001-0	3	-01--11				
5	1101110	3	-011001	2	-10-100	3	-01-1-1				
6	1011111	3	001101-	2	01010--	3	-01-11-				
6	1101111	3	0011-10	2	0101-0-	3	-011--1				
		3	-011010	2	-10100-	3	-011-1-				
		3	001110-	2	0101--0	3	-0111--				
		3	00111-0	2	-1010-0	3	-10--11				
		3	-011100	2	-101-00	3	-10-1-1				
		3	0100-11	2	10100--	3	-10-11-				
		3	010-011	2	1010-0-	3	-101--1				
		3	-100011	2	101-00-	3	-101-1-				
		3	01001-1	2	1010--0	3	-1011--				
		3	010-101	2	101-0-0	3	101---1				
		3	-100101	2	101--00	3	101--1-				
		3	010011-	2	11000--	3	101-1--				
		3	010-110	2	1100-0-	3	1011---				
		3	-100110	2	110-00-	3	110---1				
		3	01010-1	2	1100--0	3	110--1-				
		3	0101-01	2	110-0-0	3	110-1--				
		3	-101001	2	110--00	3	1101---				
		3	010101-	3	001--11						
		3	0101-10	3	-010-11						
		3	-101010	3	-01-011						
		3	010110-	3	001-1-1						
		3	01011-0	3	-0101-1						
		3	-101100	3	-01-101						
		3	10100-1	3	001-11-						
		3	1010-01	3	-01011-						
		3	101-001	3	-01-110						
		3	101001-	3	0011--1						
		3	1010-10	3	-0110-1						
		3	101-010	3	-011-01						
		3	101010-	3	0011-1-						
		3	10101-0	3	-01101-						



3	101-100	3	-011-10
3	101100-	3	00111--
3	10110-0	3	-01110-
3	1011-00	3	-0111-0
3	11000-1	3	010--11
3	1100-01	3	-100-11
3	110-001	3	-10-011
3	110001-	3	010-1-1
3	1100-10	3	-1001-1
3	110-010	3	-10-101
3	110010-	3	010-11-
3	11001-0	3	-10011-
3	110-100	3	-10-110
3	110100-	3	0101--1
3	11010-0	3	-1010-1
3	1101-00	3	-101-01
4	001-111	3	0101-1-
4	-010111	3	-10101-
4	0011-11	3	-101-10
4	-011011	3	01011--
4	00111-1	3	-10110-
4	-011101	3	-1011-0
4	001111-	3	1010--1
4	-011110	3	101-0-1
4	010-111	3	101--01
4	-100111	3	1010-1-
4	0101-11	3	101-01-
4	-101011	3	101--10
4	01011-1	3	10101--
4	-101101	3	101-10-
4	010111-	3	101-1-0
4	-101110	3	10110--
4	1010-11	3	1011-0-
4	101-011	3	1011--0
4	10101-1	3	1100--1
4	101-101	3	110-0-1
4	101011-	3	110--01
4	101-110	3	1100-1-
4	10110-1	3	110-01-
4	1011-01	3	110--10
4	101101-	3	11001--
4	1011-10	3	110-10-
4	101110-	3	110-1-0
4	10111-0	3	11010--
4	1100-11	3	1101-0-
4	110-011	3	1101--0
4	11001-1	4	-01-111
4	110-101	4	-011-11
4	110011-	4	-0111-1
4	110-110	4	-01111-
4	11010-1	4	-10-111
4	1101-01	4	-101-11
4	110101-	4	-1011-1
4	1101-10	4	-10111-
4	110110-	4	101--11
4	11011-0	4	101-1-1
5	-011111	4	101-11-
5	-101111	4	1011--1
5	101-111	4	10111-1
5	1011-11	4	10111--
5	10111-1	4	110--11
5	101111-	4	110-1-1
5	110-111	4	110-11-
5	1101-11	4	1101--1
5	11011-1	4	1101-1-
5	110111-	4	11011--
Prime Implicants			
-01----			
-10----			
Essential Prime Implicants			
-01----			
-10----			

# D<sub>0</sub>

===== Column 1 =====	===== Column 2 =====	===== Column 3 =====	===== Column 4 =====
0	0000000	0	000000-
1	0000001	0	00000-0
1	0000010	0	0-0000-
1	0000100	0	-00000-
1	0100000	0	0000--0
1	1000000	1	0-000-0
2	0000011	1	-000-00
2	0000101	1	--00000
2	0000110	1	0000--1
2	0100001	1	0-000-1
2	0100010	1	-000-01
2	0101000	1	0000-1-
2	1000001	1	0-0001-
2	1000100	1	00001--
2	1001000	1	-000100
2	1100000	1	010000-
3	0000111	1	01000-0
3	0100011	1	010-000
3	0101001	1	-100000
3	0101010	1	-10-000
3	1000101	1	1000-00
3	1001001	1	100-000
3	1001100	1	1-00000
3	1100010	2	0000-11
3	1100100	2	0-00011
3	1101000	2	00001-1
4	0101011	2	-000101
4	1001101	2	000011-
4	1100110	2	01000-1
4	1101010	2	010-001
4	1101100	2	010001-
5	1101110	2	010-010
	2	-100010	2
	2	010100-	2
	2	01010-0	2
	2	-101000	2
	2	1000-01	2
	2	100-001	2
	2	100010-	3
	2	100-100	3
	2	1-00100	3
	2	100100-	
	2	1001-00	
	2	1-01000	
	2	11000-0	
	2	1100-00	
	2	110-000	
	3	010-011	
	3	01010-1	
	3	010101-	
	3	-101010	
	3	100-101	
	3	1001-01	
	3	100110-	
	3	1-01100	
	3	1100-10	
	3	110-010	
	3	11001-0	

		3	110-100						
		3	11010-0						
		3	1101-00						
		4	110-110						
		4	1101-10						
		4	11011-0						
Prime Implicants				Essential Prime Implicants					
--00000 0000--- 0-000-- -000-0- 010-0-- -10-0-0 100--0- 1-0--00 110---0				0000--- 010-0-- 100--0- 110--0					

k-map으로 검증

D <sub>2</sub>	0000	0001	0011	0010	0110	0111	0101	0100	1100	1101	1111	1110	1010	1011	1001	1000
000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
011	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
101	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
100	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

D <sub>1</sub>	0000	0001	0011	0010	0110	0111	0101	0100	1100	1101	1111	1110	1010	1011	1001	1000
000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
110	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
101	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

D <sub>0</sub>	0000	0001	0011	0010	0110	0111	0101	0100	1100	1101	1111	1110	1010	1011	1001	1000
000	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1
110	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
100	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1

Boolean equation	$D_2 = (Q_2)'Q_1Q_0 + Q_2(Q_1)' + Q_2(Q_0)'$ $D_1 = Q_1 \oplus Q_0$ $D_0 = (Q_2)'(Q_1)'(Q_0)'(T_A)' + (Q_2)'Q_1(Q_0)'(T_{AL})'$ $+ Q_2(Q_1)'(Q_0)'(T_B)' + Q_2Q_1(Q_0)'(T_{BL})'$
------------------	---

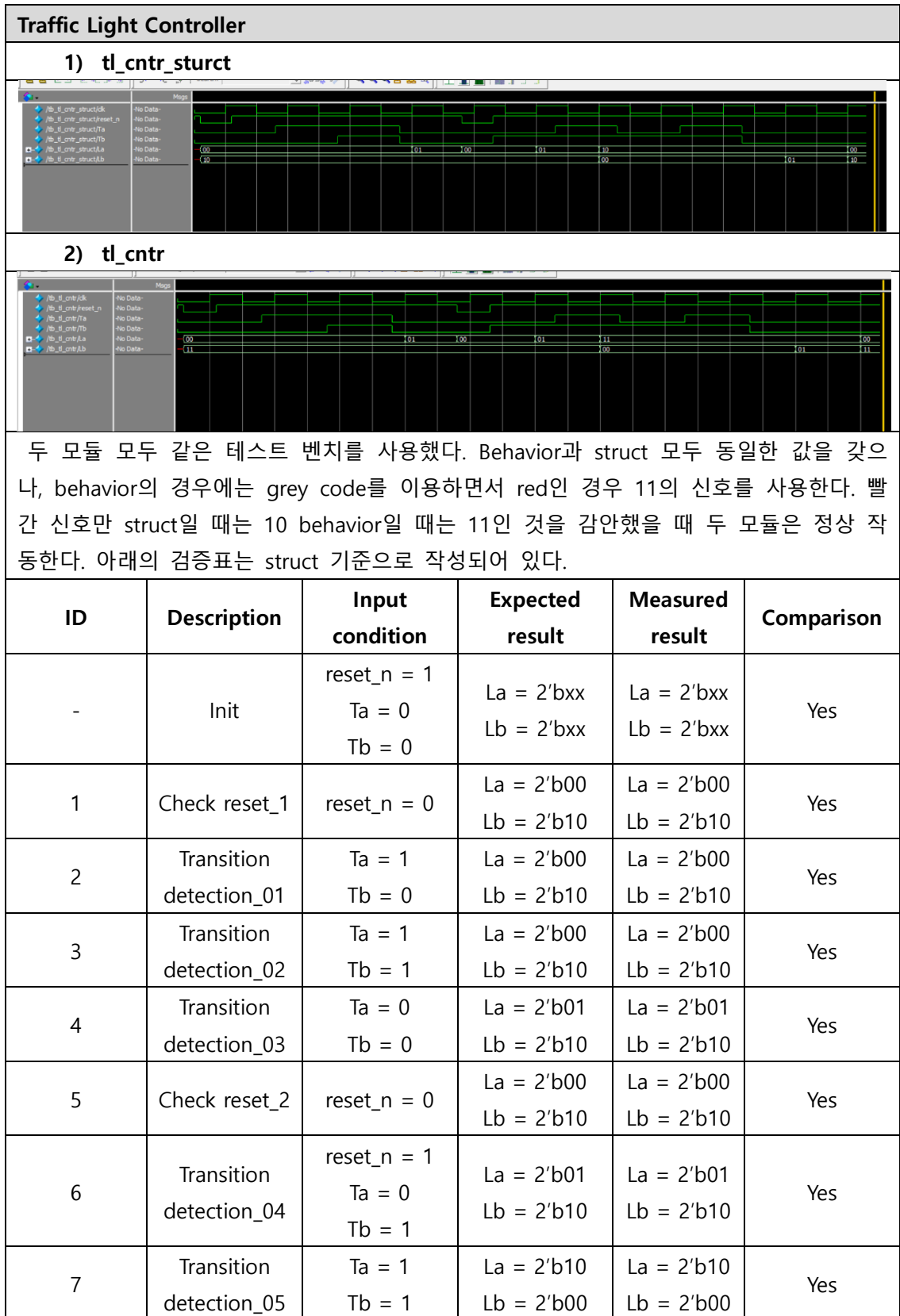
FSM output table

Current state			Output			
$Q_2$	$Q_1$	$Q_0$	$L_{A1}$	$L_{A0}$	$L_{B1}$	$L_{B0}$
0	0	0	0	0	1	1
0	0	1	0	1	1	1
0	1	0	1	0	1	1
0	1	1	0	1	1	1
1	0	0	1	1	0	0
1	0	1	1	1	0	1
1	1	0	1	1	1	0
1	1	1	1	1	0	1

Boolean equation	$L_{A1} = Q_2 + Q_1(Q_0)'$ $L_{A0} = Q_2 + Q_0$ $L_{B1} = \sim Q_2 + Q_1(Q_0)'$ $L_{B0} = \sim Q_2 + Q_0$
------------------	---

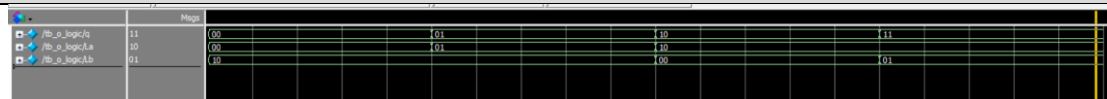
#### 4. 설계 검증 및 실험 결과

##### A. 시뮬레이션 결과



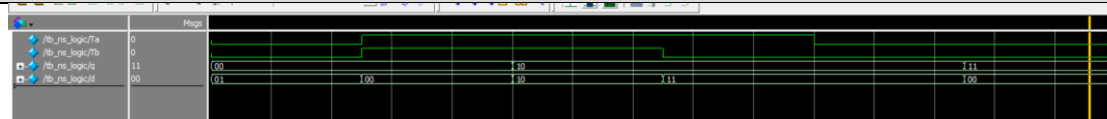
8	Transition detection_06	Ta = 0 Tb = 1	La = 2'b10 Lb = 2'b00	La = 2'b10 Lb = 2'b00	Yes
9	Transition detection_07	Ta = 1 Tb = 1	La = 2'b10 Lb = 2'b00	La = 2'b10 Lb = 2'b00	Yes
10	Transition detection_08	Ta = 0 Tb = 0	La = 2'b10 Lb = 2'b01	La = 2'b10 Lb = 2'b01	Yes
11	Transition detection_09	Ta = 0 Tb = 0	La = 2'b00 Lb = 2'b10	La = 2'b00 Lb = 2'b10	Yes

#### o\_logic



q로 들어갈 수 있는 모든 상황을 넣은 결과 총 4가지가 나오게 되고 모든 경우에서의 La, Lb값은 예상 결과대로 움직임을 알 수 있다.

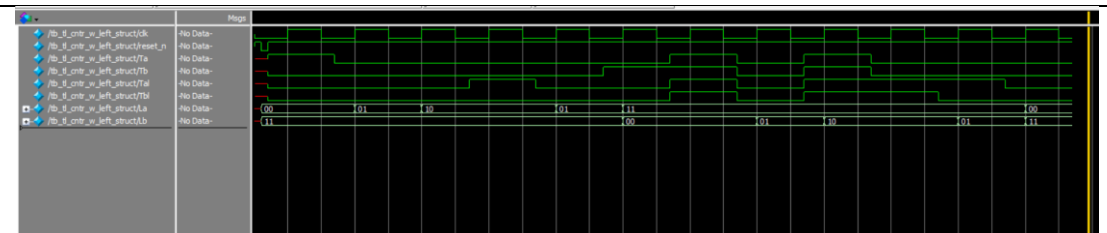
#### ns\_logic



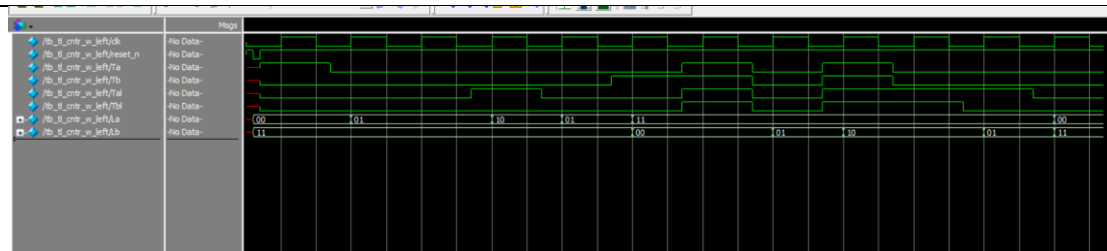
Ta, Tb는 현재 state q에 대한 input 값에 따라 d의 값이 변화하는 걸 확인할 수 있다. 의도한 대로 움직임을 알 수 있다.

### Traffic Light Controller with left

#### 1) tl\_cntr\_w\_left\_sturct



#### 2) tl\_cntr\_w\_left



두 모듈 모두 같은 테스트 벤치를 사용했다. Behavior과 struct 모두 동일한 값을 갖는 것을 확인할 수 있다.

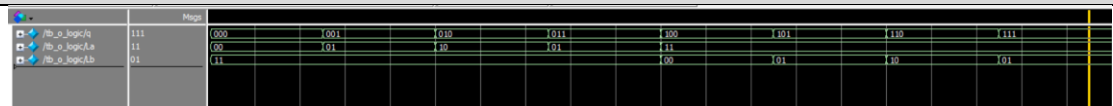
ID	Description	Input condition	Expected result	Measured result	Comparison
-	Init	reset_n = 1	La = 2'bxx	La = 2'bxx	Yes

			Lb = 2'bx	Lb = 2'bx	
1	Check reset_1	reset_n = 0	La = 2'b00 Lb = 2'b11	La = 2'b00 Lb = 2'b11	Yes
2	Transition detection_01	Ta = 1 Tb = 0 Tal = 0 Tbl = 0	La = 2'b00 Lb = 2'b11	La = 2'b00 Lb = 2'b11	Yes
3	Transition detection_02	Ta = 0 Tb = 0 Tal = 0 Tbl = 0	La = 2'b01 Lb = 2'b11	La = 2'b01 Lb = 2'b11	Yes
4	Transition detection_03	Ta = 0 Tb = 0 Tal = 0 Tbl = 0	La = 2'b01 Lb = 2'b11	La = 2'b01 Lb = 2'b11	Yes
5	Transition detection_04	Ta = 0 Tb = 0 Tal = 1 Tbl = 0	La = 2'b10 Lb = 2'b11	La = 2'b10 Lb = 2'b11	Yes
6	Transition detection_05	Ta = 0 Tb = 0 Tal = 0 Tbl = 0	La = 2'b01 Lb = 2'b11	La = 2'b01 Lb = 2'b11	Yes
7	Transition detection_06	Ta = 0 Tb = 1 Tal = 0 Tbl = 0	La = 2'b11 Lb = 2'b00	La = 2'b11 Lb = 2'b00	Yes
8	Transition detection_07	Ta = 1 Tb = 1 Tal = 1 Tbl = 1	La = 2'b11 Lb = 2'b00	La = 2'b11 Lb = 2'b00	Yes
9	Transition detection_08	Ta = 0 Tb = 0 Tal = 0 Tbl = 0	La = 2'b11 Lb = 2'b01	La = 2'b11 Lb = 2'b01	Yes
10	Transition detection_09	Ta = 1 Tb = 1 Tal = 1 Tbl = 1	La = 2'b11 Lb = 2'b10	La = 2'b11 Lb = 2'b10	Yes



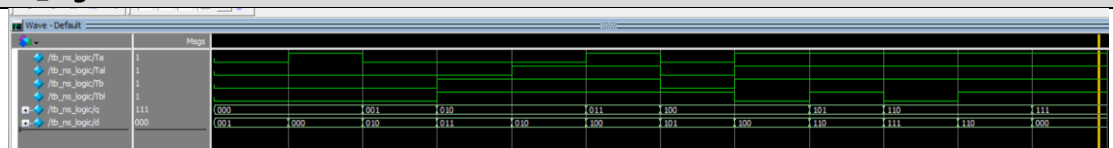
11	Transition detection_10	Ta = 0 Tb = 0 Tal = 1 Tbl = 1	La = 2'b11 Lb = 2'b10	La = 2'b11 Lb = 2'b10	Yes
12	Transition detection_11	Ta = 0 Tb = 0 Tal = 1 Tbl = 0	La = 2'b11 Lb = 2'b01	La = 2'b11 Lb = 2'b01	Yes
13	Transition detection_12	Ta = 0 Tb = 0 Tal = 0 Tbl = 0	La = 2'b00 Lb = 2'b11	La = 2'b00 Lb = 2'b11	Yes

#### o\_logic



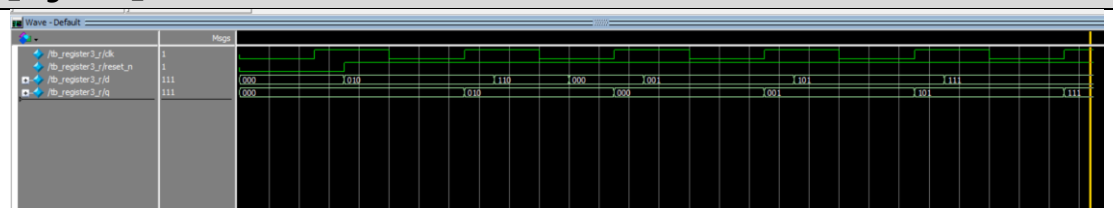
q는 3bits로 8개의 경우의 수가 나온다. 이에 대해 모든 경우의 수에 대해 La, Lb의 값을 확인한다. 각 state(q의 값)에 따라 La, Lb가 잘 나오는 것을 확인할 수가 있다.

#### ns\_logic



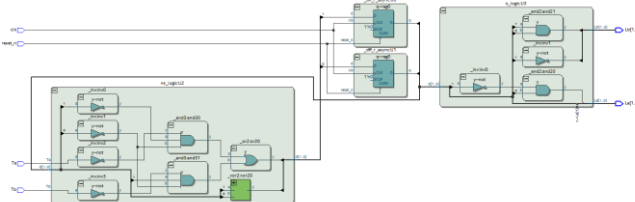
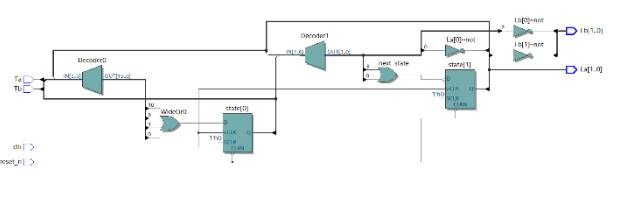
입력값은 3bits인 q와 Ta, Tb, Tal, Tbl을 이용하면 결과로 3bits d가 나온다. 각 state에 따른 입력값에 따라 다음 state를 가는지 결정한다. State가 encoding이 짝수인 경우에는 next state로 가는 경우엔 알맞은 입력값으로 Ta, Tb, Tal, Tbl을 주었을 경우이다. 하지만 홀수인 경우에는 신호가 yellow와 left신호를 켜지고 나서이다. 해당 state에서 다음 state로 갈때에는 어떤 값이던 되고 있는 것을 볼 수 있다.

#### \_register3\_r



해당 테스트벤치는 \_register3\_r로 resettable register를 3개를 연결해서 사용한다. 처음에는 reset\_n이 하강하면서 q의 값은 000으로 결정된다. reset\_n이 1인 경우에는 clk이 상승할 때마다 q의 값이 update되는 것을 확인할 수 있다. clk이 상승하지 않을 때의 변한 값은 q에 반영되지 않는 것을 d의 값이 3'b110일 때 확인할 수 있다.

## B. 합성(synthesis) 결과

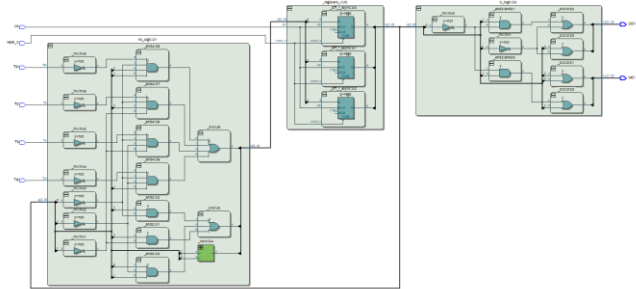
Traffic Light Controller																																							
1) tl_cntr_struct																																							
	<div>Flow Summary</div> <div>&lt;&lt;Filter&gt;&gt;</div> <table><tr><td>Flow Status</td><td>Successful - Wed Oct 11 20:39:02 2023</td></tr><tr><td>Quartus Prime Version</td><td>18.1.0 Build 625 09/12/2018 SJ Lite Edition</td></tr><tr><td>Revision Name</td><td>tl_cntr_struct</td></tr><tr><td>Top-level Entity Name</td><td>tl_cntr_struct</td></tr><tr><td>Family</td><td>Cyclone V</td></tr><tr><td>Device</td><td>5CSXFC6D6F31C6</td></tr><tr><td>Timing Models</td><td>Final</td></tr><tr><td>Logic utilization (in ALMs)</td><td>3 / 41,910 (&lt; 1 %)</td></tr><tr><td>Total registers</td><td>3</td></tr><tr><td>Total pins</td><td>8 / 499 (2 %)</td></tr><tr><td>Total virtual pins</td><td>0</td></tr><tr><td>Total block memory bits</td><td>0 / 5,662,720 (0 %)</td></tr><tr><td>Total DSP Blocks</td><td>0 / 112 (0 %)</td></tr><tr><td>Total HSSI RX PCSs</td><td>0 / 9 (0 %)</td></tr><tr><td>Total HSSI PMA RX Deserializers</td><td>0 / 9 (0 %)</td></tr><tr><td>Total HSSI TX PCSs</td><td>0 / 9 (0 %)</td></tr><tr><td>Total HSSI PMA TX Serializers</td><td>0 / 9 (0 %)</td></tr><tr><td>Total PLLs</td><td>0 / 15 (0 %)</td></tr><tr><td>Total DLLs</td><td>0 / 4 (0 %)</td></tr></table>	Flow Status	Successful - Wed Oct 11 20:39:02 2023	Quartus Prime Version	18.1.0 Build 625 09/12/2018 SJ Lite Edition	Revision Name	tl_cntr_struct	Top-level Entity Name	tl_cntr_struct	Family	Cyclone V	Device	5CSXFC6D6F31C6	Timing Models	Final	Logic utilization (in ALMs)	3 / 41,910 (< 1 %)	Total registers	3	Total pins	8 / 499 (2 %)	Total virtual pins	0	Total block memory bits	0 / 5,662,720 (0 %)	Total DSP Blocks	0 / 112 (0 %)	Total HSSI RX PCSs	0 / 9 (0 %)	Total HSSI PMA RX Deserializers	0 / 9 (0 %)	Total HSSI TX PCSs	0 / 9 (0 %)	Total HSSI PMA TX Serializers	0 / 9 (0 %)	Total PLLs	0 / 15 (0 %)	Total DLLs	0 / 4 (0 %)
Flow Status	Successful - Wed Oct 11 20:39:02 2023																																						
Quartus Prime Version	18.1.0 Build 625 09/12/2018 SJ Lite Edition																																						
Revision Name	tl_cntr_struct																																						
Top-level Entity Name	tl_cntr_struct																																						
Family	Cyclone V																																						
Device	5CSXFC6D6F31C6																																						
Timing Models	Final																																						
Logic utilization (in ALMs)	3 / 41,910 (< 1 %)																																						
Total registers	3																																						
Total pins	8 / 499 (2 %)																																						
Total virtual pins	0																																						
Total block memory bits	0 / 5,662,720 (0 %)																																						
Total DSP Blocks	0 / 112 (0 %)																																						
Total HSSI RX PCSs	0 / 9 (0 %)																																						
Total HSSI PMA RX Deserializers	0 / 9 (0 %)																																						
Total HSSI TX PCSs	0 / 9 (0 %)																																						
Total HSSI PMA TX Serializers	0 / 9 (0 %)																																						
Total PLLs	0 / 15 (0 %)																																						
Total DLLs	0 / 4 (0 %)																																						
<p>현재 ns_logic과 _dff_r_async, o_logic으로 RTL이 구성된 것을 확인할 수 있다. ns_logic에는 q와 Ta, Tb의 조합으로 이루어진 것을 확인할 수 있으며 L1과 L0는 q의 조합으로만 이루어진 것을 확인할 수 있다. 구성을 확인하면 sequential circuits 부분과 combinational circuits 부분이 나누어진 것을 확인할 수 있다.</p>																																							
2) tl_cntr																																							
	<div>Flow Summary</div> <div>&lt;&lt;Filter&gt;&gt;</div> <table><tr><td>Flow Status</td><td>Successful - Sat Oct 14 18:30:24 2023</td></tr><tr><td>Quartus Prime Version</td><td>18.1.0 Build 625 09/12/2018 SJ Lite Edition</td></tr><tr><td>Revision Name</td><td>tl_cntr</td></tr><tr><td>Top-level Entity Name</td><td>tl_cntr</td></tr><tr><td>Family</td><td>Cyclone V</td></tr><tr><td>Device</td><td>5CGXFC7C6U19C6</td></tr><tr><td>Timing Models</td><td>Final</td></tr><tr><td>Logic utilization (in ALMs)</td><td>3 / 56,480 (&lt; 1 %)</td></tr><tr><td>Total registers</td><td>3</td></tr><tr><td>Total pins</td><td>8 / 268 (3 %)</td></tr><tr><td>Total virtual pins</td><td>0</td></tr><tr><td>Total block memory bits</td><td>0 / 7,024,640 (0 %)</td></tr><tr><td>Total DSP Blocks</td><td>0 / 156 (0 %)</td></tr><tr><td>Total HSSI RX PCSs</td><td>0 / 6 (0 %)</td></tr><tr><td>Total HSSI PMA RX Deserializers</td><td>0 / 6 (0 %)</td></tr><tr><td>Total HSSI TX PCSs</td><td>0 / 6 (0 %)</td></tr><tr><td>Total HSSI PMA TX Serializers</td><td>0 / 6 (0 %)</td></tr><tr><td>Total PLLs</td><td>0 / 13 (0 %)</td></tr><tr><td>Total DLLs</td><td>0 / 4 (0 %)</td></tr></table>	Flow Status	Successful - Sat Oct 14 18:30:24 2023	Quartus Prime Version	18.1.0 Build 625 09/12/2018 SJ Lite Edition	Revision Name	tl_cntr	Top-level Entity Name	tl_cntr	Family	Cyclone V	Device	5CGXFC7C6U19C6	Timing Models	Final	Logic utilization (in ALMs)	3 / 56,480 (< 1 %)	Total registers	3	Total pins	8 / 268 (3 %)	Total virtual pins	0	Total block memory bits	0 / 7,024,640 (0 %)	Total DSP Blocks	0 / 156 (0 %)	Total HSSI RX PCSs	0 / 6 (0 %)	Total HSSI PMA RX Deserializers	0 / 6 (0 %)	Total HSSI TX PCSs	0 / 6 (0 %)	Total HSSI PMA TX Serializers	0 / 6 (0 %)	Total PLLs	0 / 13 (0 %)	Total DLLs	0 / 4 (0 %)
Flow Status	Successful - Sat Oct 14 18:30:24 2023																																						
Quartus Prime Version	18.1.0 Build 625 09/12/2018 SJ Lite Edition																																						
Revision Name	tl_cntr																																						
Top-level Entity Name	tl_cntr																																						
Family	Cyclone V																																						
Device	5CGXFC7C6U19C6																																						
Timing Models	Final																																						
Logic utilization (in ALMs)	3 / 56,480 (< 1 %)																																						
Total registers	3																																						
Total pins	8 / 268 (3 %)																																						
Total virtual pins	0																																						
Total block memory bits	0 / 7,024,640 (0 %)																																						
Total DSP Blocks	0 / 156 (0 %)																																						
Total HSSI RX PCSs	0 / 6 (0 %)																																						
Total HSSI PMA RX Deserializers	0 / 6 (0 %)																																						
Total HSSI TX PCSs	0 / 6 (0 %)																																						
Total HSSI PMA TX Serializers	0 / 6 (0 %)																																						
Total PLLs	0 / 13 (0 %)																																						
Total DLLs	0 / 4 (0 %)																																						
<p>해당 RTL map은 tl_cntr을 behavior로 구현했을 때 나오는 모습이다. 2개의 register로 이루어진 모습과 decoder를 이용한 모습을 확인할 수 있다.</p>																																							
<p>Top module은 tl_cntr이고, Logic Utilization은 3, register는 3개이다. total pins는 8개이다. Flow status를 보았을 때 successful로 syntax error가 없는 것을 확인할 수 있다.</p>																																							
<p>Top module은 tl_cntr이고, Logic Utilization은 3, register는 3개이다. total pins는 8개이다. Flow status를 보았을 때 successful로 syntax error가 없는 것을 확인할 수 있다.</p>																																							

Top module은 tl\_cntr\_struct이고, Logic Utilization은 3, register는 3개이다. total pins는 8개이다. Flow status를 보았을 때 successful로 syntax error가 없는 것을 확인할 수 있다.

Top module은 tl\_cntr이고, Logic Utilization은 3, register는 3개이다. total pins는 8개이다. Flow status를 보았을 때 successful로 syntax error가 없는 것을 확인할 수 있다.

## Traffic Light Controller with left turn

### 1) tl\_cntr\_w\_left\_struct



#### Flow Summary

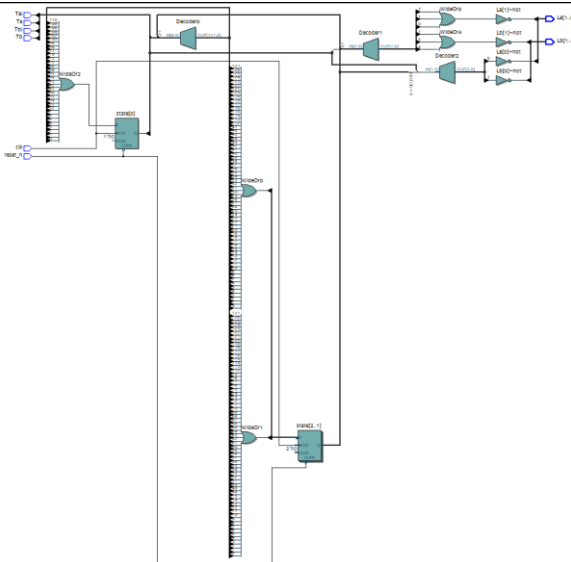
<<Filter>>

Flow Status	Successful - Wed Oct 11 22:20:17 2023
Quartus Prime Version	18.1.0 Build 625 09/12/2018 SJ Lite Edition
Revision Name	tl_cntr_w_left_struct
Top-level Entity Name	tl_cntr_w_left_struct
Family	Cyclone V
Device	5C5XFC6D6F31C6
Timing Models	Final
Logic utilization (in ALMs)	5 / 41,910 (< 1 %)
Total registers	4
Total pins	10 / 499 (2 %)
Total virtual pins	0
Total block memory bits	0 / 5,662,720 (0 %)
Total DSP Blocks	0 / 112 (0 %)
Total HSSI RX PCSs	0 / 9 (0 %)
Total HSSI PMA RX Deserializers	0 / 9 (0 %)
Total HSSI TX PCSs	0 / 9 (0 %)
Total HSSI PMA TX Serializers	0 / 9 (0 %)
Total PLLs	0 / 15 (0 %)
Total DLLs	0 / 4 (0 %)

현재 ns\_logic과 \_dff\_r\_async, o\_logic으로 RTL이 구성된 것을 확인할 수 있다. ns\_logic에는 q와 Ta, Tb, Tal, Tbl의 조합으로 이루어진 것을 확인할 수 있으며 L1과 L0는 q의 조합으로만 이루어진 것을 확인할 수 있다. 구성을 확인하면 sequential circuits 부분과 combinational circuits 부분이 나누어진 것을 확인할 수 있다.

Top module은 tl\_cntr\_w\_left\_struct이고, Logic Utilization은 5, register는 4개이다. total pins는 10개이다. Flow status를 보았을 때 successful로 syntax error가 없는 것을 확인할 수 있다.

### 2) tl\_cntr\_w\_left



#### Flow Summary

<<Filter>>

Flow Status	Successful - Sat Oct 14 18:47:08 2023
Quartus Prime Version	18.1.0 Build 625 09/12/2018 SJ Lite Edition
Revision Name	tl_cntr_w_left
Top-level Entity Name	tl_cntr_w_left
Family	Cyclone V
Device	5C5XFC6D6F31C6
Timing Models	Final
Logic utilization (in ALMs)	5 / 41,910 (< 1 %)
Total registers	4
Total pins	10 / 499 (2 %)
Total virtual pins	0
Total block memory bits	0 / 5,662,720 (0 %)
Total DSP Blocks	0 / 112 (0 %)
Total HSSI RX PCSs	0 / 9 (0 %)
Total HSSI PMA RX Deserializers	0 / 9 (0 %)
Total HSSI TX PCSs	0 / 9 (0 %)
Total HSSI PMA TX Serializers	0 / 9 (0 %)
Total PLLs	0 / 15 (0 %)
Total DLLs	0 / 4 (0 %)

해당 RTL map은 tl\_cntr\_w\_left를 behavior로 구현했을 때 나오는 모습이다. 2개의 register로 이루어진 모습과 decoder를 이용한 모습을 확인할 수 있다.

Top module은 tl\_cntr\_w\_left이고, Logic Utilization은 5, register는 4개이다. total pins는 10개이다. Flow status를 보았을 때 successful로 syntax error가 없는 것을 확인할 수 있다.

## 5. 고찰 및 결론

### A. 고찰

Quine Mccluskey method을 이용하여  $D_2$ ,  $D_1$ ,  $D_0$ 의 부울식을 구할 때, 예시로 보여준  $D_2$ 에서부터 생각보다 너무 많은 column이 나와 비교하기가 쉽지 않았다. 이에 대한 방법을 강구하던 도중 1학기 디지털논리회로1에서 구현했던 qm 알고리즘을 이용하여 각 column과  $p_i$ ,  $ep_i$ 를 구할 수 있었다. 이 프로그램을 활용하여 traffic light controller with left struct의 next state에 대한 식을 어렵지 않게 구할 수 있었다.

### B. 결론

Behavior로 코드 짜는 거 어렵지 않다. 오히려 gate를 이용해서 구하는 것보다 굉장히 시간도 단축할 수 있었던 코딩 스타일이었다. 하지만, 이를 이용하면 하드웨어 설계에서 어떤 방식으로 해당 값을 도출할 수 있었는지 알 수 없기 때문에 struct 스타일의 방식이 중요하다는 것을 깨달았다. Behavior는 struct를 짜기 전 전체적인 프로그램을 미리 결과를 테스트하는 용도이지 않을까 생각해 보았다.

## 6. 참고문헌

유지현 교수님/디지털논리회로1/광운대학교(컴퓨터정보공학부)/2023

이준환 교수님/컴퓨터공학기초실험/광운대학교(컴퓨터정보공학부)/2023