

# 컴퓨터 공학 기초 설계 및 실험 2 – Lab 7

## Traffic Light Controller

이번 실습에서는 FSM의 기법 중 하나인 Moore FSM을 적용하여 traffic light controller를 설계한다.

### 1. Traffic Light Controller

#### A. Description

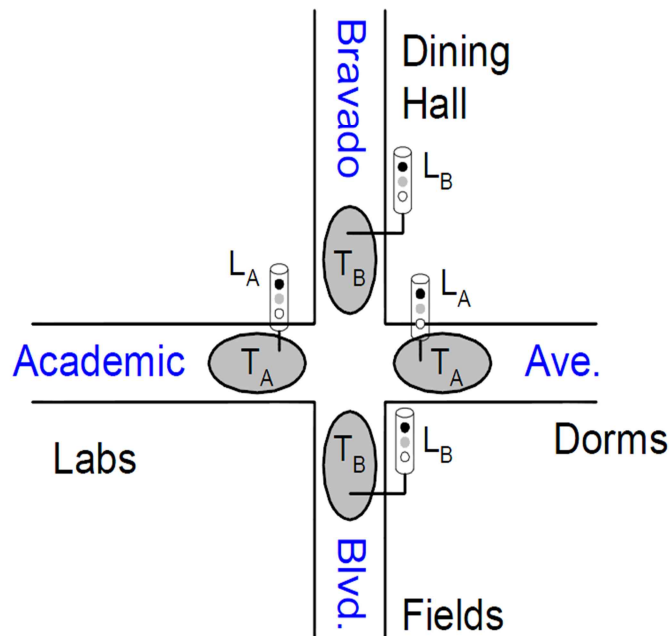


Figure 1 - Traffic light controller

이번 실습에서는 위의 그림 1과 같은 신호등을 제어하는 logic을 구현한다. 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 설치하였다.

신호등의 구현을 위해 다음 규칙을 만족해야 한다.

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

## B. Structural description

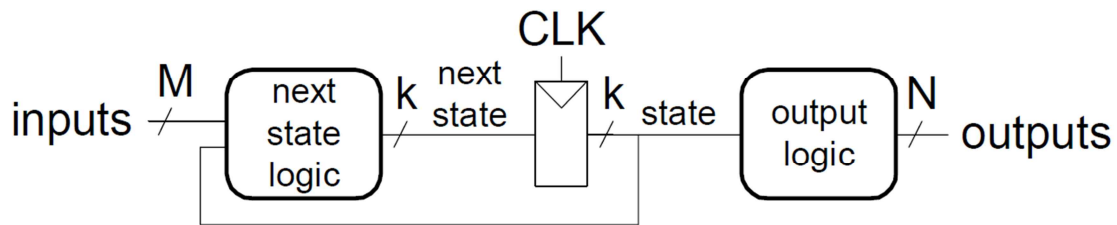


Figure 2 - Moore FSM

이번에 구현할 traffic light controller는 그림 2의 Moore FSM처럼 구현하여 본다.

## 2. 실습

### A. Design 단계

- i. 앞서 Lab 6에서 하였던 것처럼 sequential logic 설계 단계에 맞추어서 구현하도록 한다.

(각각의 설계 단계에 대하여 report에 작성하고 이에 대하여 설명한다.)

- ii. Sequential logic의 설계 단계는 다음과 같다.

#### (1) Drawing the finite state diagram

(1-1) Define states

(1-2) Define inputs

(1-3) Define outputs

(1-4) Draw the diagram

**Diagram 작성 후 state transition table과 output table을 작성하고, 이에 대하여 Boolean equation으로 report에 정리한다.**

(2) Encoding states

(3) Coding the module header

(4) Coding state registers(flip-flops) – sequential circuits

(5) Coding combinational circuits

## B. Implementation

Traffic light controller를 구현할 때, 구조적으로 구현한다. 다음의 표는 최소한으로 포함하여야 하는 module configuration을 나타낸다.

(이들을 report에 정리하고, 또한 추가적인 모듈이 필요하다면 추가한 후 이에 대한 기능을 설명하도록 한다.)

구분	이름	설명
<b>Top module</b>	tl_cntr	Traffic light controller의 top module
<b>Sub module</b>	ns_logic	Traffic light controller의 next state를 결정하는 combinational logic
<b>Sub module</b>	_register2_r_async	2-bit resettable register with active low asynchronous reset module (내부에 d_ff_r_async를 instance) - 현재 state의 값을 저장하고 있다.
<b>Sub module</b>	_dff_r_async	Resettable D flip-flop with active low asynchronous reset
<b>Sub module</b>	o_logic	현재 state의 값에 기반하여 output 값을 결정하는 combinational logic

**Table 1 - Module configuration**

(기타 필요한 gate들에 대하여서는 module을 추가하여도 되고 assign 구문으로 처리하여도 무관하다.)

다음은 top module의 I/O configuration을 나타낸다. Top module을 제외한 port의 이름과 비트는 자유롭게 해도 무관하다.

Module 이름	구분	이름	비트 수	설명
<b>tl_cntr</b>	input	clk	1-bit	Clock
		reset_n	1-bit	Active low에 동작하는 reset 신호로 state를 초기화
		Ta	1-bit	Traffic sensor A('Academic Ave.'에 위치)
		Tb	1-bit	Traffic sensor B('Bravado Blvd.'에 위치)
	output	La	2-bit	신호등 값 출력 A('Academic Ave.'에 위치)
		Lb	2-bit	신호등 값 출력 B('Bravado Blvd.'에 위치)

**Table 2 – I/O configuration of top module**

### 3. Report

Report 는 다음의 항목들에 대하여 정리하도록 한다.

- (1) Traffic light controller
- (2) Report는 다음과 같은 내용을 반드시 포함하여 작성한다.
  - 1) Assignment Specification
  - 2) Design Details
  - 3) Design Verification and Results
  - 4) Discussion and Conclusion

(이번 실습에서는 technology map viewer는 포함하지 않는다.  
Flow summary와 RTL viewer는 포함한다.)

### 4. Submission

- (1) Soft copy
  - A. 월요일 실습 - 11 월 06 일 23 시 59 분
  - B. 화요일 실습 - 11 월 07 일 23 시 59 분
  - C. 수요일 실습 - 11 월 08 일 23 시 59 분
  - D. 실습 미수강 - 11월 08일 23시 59분까지

**Source code와 hard copy를 같이 압축하여 u-Campus(종합정보서비스) 과제 제출에 해당 과제 upload (실습 미수강 인원은 [ccw0604@gmail.com](mailto:ccw0604@gmail.com)으로 보낼 것)**

- (2) Hard copy
  - A. 다음날 실습시간, 이론만 듣는 학생은 11 월 09 일 수업시간에 제출
  - B. 늦은 숙제는 다음 실험 시작 전까지만 받음(20% 감점)

## 5. 채점기준

세부사항		점수	최상	상	중	하	최하
소스 코드	Source code가 잘 작성되었는가? (구조적으로 작성하였는가?)	15	15	10	5		0
	주석을 적절히 달았는가?	5	5				0
	입/출력port들이 design specification에 맞춰 작성되었는가?	10	10		5		0
설계 검증	Waveform이 정확한가?	10	10	8	5	3	0
	Waveform에 대한 분석이 이루어졌는가?	20	20	15	10	5	0
	Synthesis 결과에 대해 설명하였는가?	10	10	8	5	3	0
	적절한 입력조합에 대해 검증되었는가?	30	30	22	14	6	0
총점		100					