

# REPORT



제목 : 9조 텀프로젝트 결과 보고서

수강과목 : 논리회로설계및실험

담당교수 : 권동현 교수

학 과 : 정보컴퓨터공학부

학 번 : 202255668, 202255663

이 름 : 이승원, 설종환

제출일자 : 2024.12.24

# 목차

1. 목적
2. 시나리오
3. 개발내용
4. 기존과 달라진 개발 내용
5. 모듈 기능 요약

## 1. 목적

이 프로젝트는 LED가 특정 패턴으로 점등되는 순서를 플레이어가 기억하여 키패드로 입력하는 게임을 구현하는 데 목적이 있다. 게임은 다음과 같은 과정을 따른다. 모듈 내부에 LED 점등 패턴이 저장되어있고, 이 패턴이 LED가 순차적으로 점등됨으로써 표시된다. 플레이어는 LED 점등 순서를 외운 후 키패드를 이용해 입력한다.

플레이어가 정답을 정확히 입력할 경우, full-color LED가 초록색으로 점등되며 게임이 종료되었음을 알린다. 이 게임은 플레이어의 기억력과 순발력을 테스트하는 간단하고 직관적인 엔터테인먼트 경험을 제공하는 데 중점을 둔다.

## 2. 시나리오

게임을 시작하면 타이머가 활성화되고 점수를 결정하기 위한 시간이 카운트된다. 점수는 9점에서 시작하며 10초가 지날 때마다 1점씩 떨어지는 방식이다. 내부에 저장한 패턴에 따라 led가 순서대로 점등되고 나면 플레이어는 패턴을 입력한다. 이때 키패드를 누르면 순서대로 array 세그먼트 디스플레이에 숫자가 표시되며 입력을 마치면 코드 내부에 저장되어 있는 패턴값과 비교한다. 일치할 경우 full color led는 초록색으로 점등, led는 패턴대로 점등하던 것을 멈추고 8개의 모두 점등되며, 점수가 더 이상 떨어지지 않고 멈추며 게임이 종료된다.

## 3. 개발내용

여러 모듈을 개별적으로 개발하고 상위 파일 week11.v로 합쳐서 전체적인 작동을 하는 방식으로 개발했다. 일부는 verilog코드로 작성했고 일부는 quartus 앱으로 직접 회로를 구현했다. 개발한 모듈이 어떤 방식으로 작동하고 전체 프로젝트에 어떤 영향을 주는지 설명하겠다.

### 1. 최상위 모듈(week11.v)

최상위 모듈인 week11은 LED 패턴 매칭 게임의 핵심 컨트롤러로, 모든 서브 모듈을 통합하여 시스템 전반의 동작을 제어한다. 먼저, 플레이어가 키패드로 입력한 값은 trigger 모듈에서 디바운싱 처리된 후 d2b 모듈을 통해 4비트 이진수로 변환된다. 해당 값은 4비트 레지스터가 8개가 순차적으로 연결되어 쉬프트 레지스터로 작동하는 eight\_digit\_shift\_register 모듈에 저장된다.

입력된 값은 Password 모듈에서 사전에 설정된 LED 점등 패턴과 비교되어 정답 여부를 판별한다. 정답일 경우, Correct 신호가 활성화되어 키패드 입력이 더 이상 처리되지 않고 8개의 모든 led가 점등되며, LED 모듈을 통해 Full-Color LED가 초록색으로 점등된다.

정답 입력 전에는 simon 모듈이 활성화되어 사전 정의된 LED 점등 패턴을 반복적으로 표시한다. 정답 입력 시, timer 모듈이 측정한 시간 값을 바탕으로 score\_calculator 모듈에서 점수가 산출되며, 점수는 정답 입력 시간에 따라 9점에서 1점 사이로 부여된다. 이 점수는 single\_seg\_decoder 모듈을 통해 단일 7-세그먼트 디스플레이에 표시된다. 키패드 입력은 b2seg\_bus와 display\_controller 모듈을 통해 입력된 패턴을 배열 7-세그먼트 디스플레이에 출력하여 플레이어가 입력한 값을 확인할 수 있도록 한다.

이 모든 과정에서 각 모듈은 PNU\_CLK\_DIV 모듈에서 분주된 클럭 신호를 사용하여 동작하며, count\_8 모듈과 display\_controller 모듈은 8개의 7-세그먼트 디스플레이가 간상 효과를 통해 매끄럽게 동작하도록 한다.

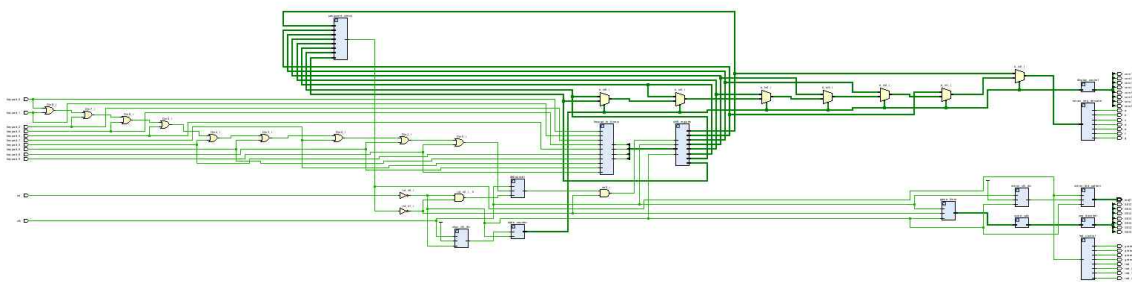


그림-1. week11 모듈 심볼

## 2. trigger 모듈(trigger.v)

이 모듈은 키패드를 입력했을 때 디바운싱이 발생하는 것을 방지하기 위해 사용한다. 키패드의 모든 입력 신호를 하나로 결합하여 디바운싱 후 debounced\_key로 안정화된 신호를 생성하는 역할을 한다.

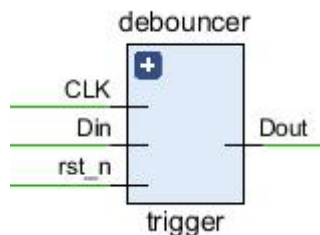


그림-2. trigger 모듈 심볼

## 3. keypad to binary conversion 모듈(d2b.v)

이 모듈은 키패드의 입력 신호를 4비트 이진수 값으로 변환하는 역할을 한다. 레지스터에 저장하여 나중에 password와 비교하기 위해 이진수로 변환하는 과정을 거쳐야 하므로 사용한다.

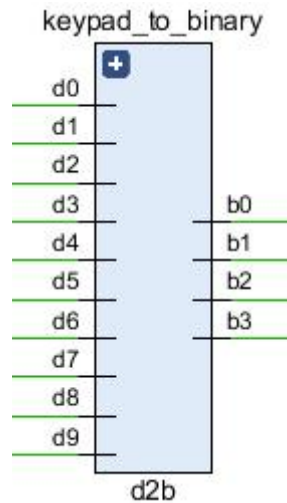


그림-3. d2b 모듈 심볼

#### 4. 시프트 레지스터 모듈(eight\_digit\_shift\_register.v)

이 모듈은 8개의 4비트 데이터 레지스터를 직렬로 연결해서 4비트 데이터를 순차적으로 이동시키는 시프트 레지스터이다. 키패드로 패턴을 입력할 때 숫자가 입력될 때마다 이전 입력 값이 옆으로 밀려야 패턴 순서대로 저장이 되기 때문에 사용한다.

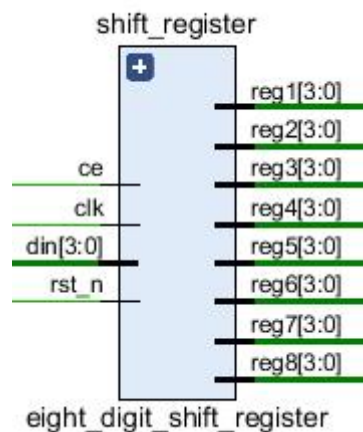


그림-4. eight\_digit\_shift\_register 모듈 심볼

#### 5. 4비트 레지스터 모듈(four\_bit\_reg\_ce.v)

이 모듈은 키패드로 입력되는 숫자를 이진수로 변환하여 저장하기 위해 사용한다. 키패드 숫자를 변환하면 4비트까지 나오기 때문에 4비트로 구현했으며 이 모듈을 사용하여 위의 eight\_digit\_shift\_register 모듈을 구현했다.

#### 6. password 모듈(Password.v)

이 모듈은 입력 받은 8개의 4비트 데이터를 미리 저장해놓은 패턴 8자리와 일치하는

지 비교하여 결과를 출력하는 모듈이다. 일치할 경우 correct 신호를 1로 활성화 시킨다. 입력 받은 패턴을 저장된 패턴과 비교하기 위해 사용한다.

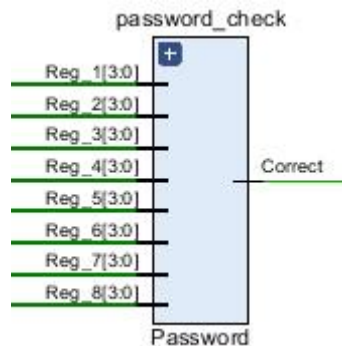


그림-5. password 모듈 심볼

## 7. full-color-led 모듈(LED.v)

이 모듈은 password 모듈에서 비밀번호를 비교했을 때 결과가 일치할 경우 활성화 되는 correct 신호 1을 받았을 때 초록색으로 활성화된다. 플레이어가 게임에서 이겼음을 확인하기 위해 사용한다.

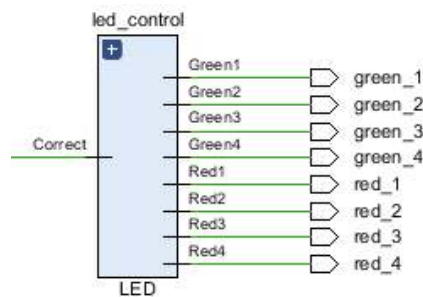


그림-6. LED 모듈 심볼

## 8. binary to segment 모듈(b2seg\_bus.v)

이 모듈은 4비트 이진수 입력을 받아서 7-세그먼트 디스플레이의 a~g로 변환하는 역할을 한다. 내부적으로 이진수 입력을 기반으로 각 세그먼트의 활성화 상태를 결정하는 논리 연산을 수행하며, 입력 값에 따라 1부터 9까지의 숫자를 정확히 출력하도록 하기 위해 사용한다.

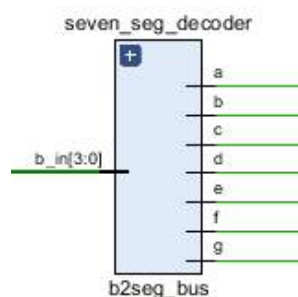


그림-7. b2seg\_bus 모듈 심볼

## 9. display-controller 모듈(display-controller.v)

3비트 입력 신호(count)를 받아 8비트 출력 신호(com)를 제어하여, 8개의 7-Segment 디스플레이 중 하나만 활성화한다. 8진 카운터의 적절한 클럭 분주를 통해 잔상효과로 디스플레이 8자리에 키패드 입력 값을 원활하게 보이게 한다.

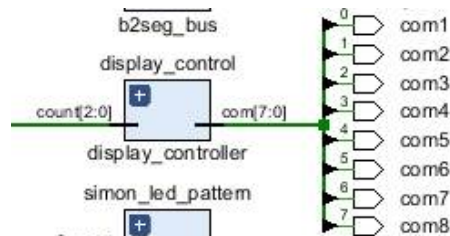


그림-8. display\_controller 모듈 심볼

## 10. 8진 카운터(count\_8.v)

이 모듈은 8진 카운터로 com1~com8이 하나씩 순서대로 활성화되게 하는 역할을 한다. 세그먼트 디스플레이를 활성화시키기 위해 사용되며 클럭을 1MHz로 할당하면 com1~com8로 돌아가는 순서가 빨라져서 잔상효과로 인해 com1~com8이 모두 켜진 것 같은 효과를 낸다.

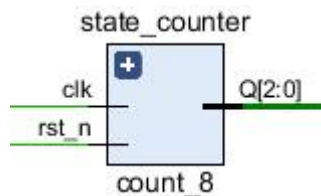


그림-9. count\_8 모듈 심볼

## 11. 클럭 분주기(PNU\_CLK\_DIV.v)

이 모듈은 모듈 별로 클럭을 분주시켜줘야 하는 값이 다르기 때문에 각자 클럭을 할당해주기 위해 사용한다. 입력 클럭(clk) 신호를 분주하여 지정된 주파수의 클럭 신호(div\_clk)를 생성한다. 활성화 신호(en)가 켜져 있는 동안 동작하며, 리셋 신호(rst\_n)가 낮을 경우 카운터(cnt)와 분주 클럭(div\_clk)을 초기화한다.

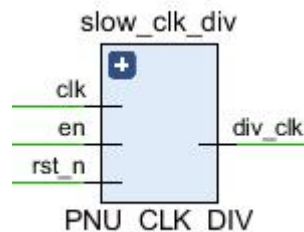


그림-10. PNU\_CLK\_DIV 모듈 심볼

## 12. led 패턴 점등 모듈(simon.v)

이 모듈은 8개의 LED를 순서대로 점등하며 플레이어에게 비밀번호 패턴을 시각적으로 보여주기 위해 사용한다. 내부적으로는 pattern 배열에 LED 점등 패턴이 저장되어 있으며, pattern\_index를 사용해 현재 점등 중인 LED를 가리킨다. blink\_state는 LED 상태를 OFF(0), ON(1), NEXT(2)로 나누어 점등 과정을 제어한다.

리셋 신호(rst\_n)가 비활성화되면 모든 LED를 OFF 상태로 초기화하고, pattern\_index와 blink\_state를 초기값으로 설정한다. 정답 신호(Correct)가 활성화되면 active 플래그를 비활성화 하여 점등 동작을 중단하고 모든 LED를 점등 시킨다.

정답이 맞춰지지 않은 경우(Correct가 비활성화)에는 blink\_state를 기반으로 현재 LED를 OFF에서 ON으로 전환하며, 점등이 완료되면 pattern\_index를 0~7 사이에서 순환하도록 증가시켜 다음 LED를 점등시킨다. 이를 통해 패턴이 지속적으로 출력되며, clk 입력 신호에 따라 LED의 점등 간격이 조절될 수 있다.

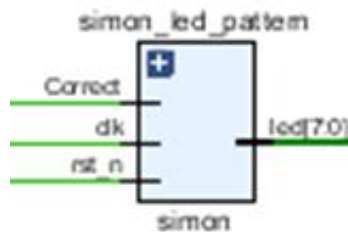


그림-11. simon 모듈 심볼

## 13. 싱글 세그먼트 디코더(single\_seg\_decoder.v)

이 모듈은 4비트 이진수 입력값을 받아 a~g로 디코딩하여 1자리 세그먼트 디스플레이에 출력하는 역할을 한다. 시간에 따라 달라지는 점수를 표시하기 위해 사용한다.

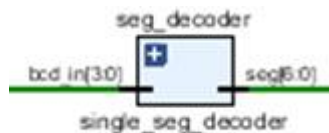


그림-12. single\_seg\_decoder 모듈 심볼

## 14. 점수 계산 모듈(score\_calculator.v)

이 모듈은 time\_count 값을 입력받아 시간에 따라 점수를 계산하여 출력하는 역할을 수행한다. 내부 로직은 always @(\*) 블록을 사용하여 입력된 time\_count 값을 조건문으로 평가하고, 이에 따라 score 값(9~1)을 결정한다.

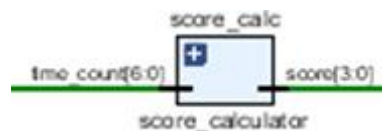


그림-13. score\_calculator 모듈 심볼



## 15. 타이머 모듈(timer.v)

점수 계산 모듈에서 시간을 카운트하기 위해 사용한다. 주어진 클럭 신호와 활성화 신호를 기반으로 초 단위 시간을 카운트하여 time\_count로 출력하는 역할을 수행한다. 1MHz의 클럭을 통해 1초 간격의 카운트를 생성하며, 최대 127초까지 측정 가능하다.

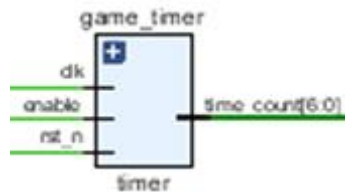


그림-14. timer 모듈 심볼

## 4. 기존계획과 달라진 개발내용

처음에는 다른 사용자가 키패드를 통해 패턴을 입력하고 그 후에 플레이어가 패턴을 입력하는 방식으로 게임을 진행하려 했으나, 구현에 실패해서 내부에서 패턴을 직접 지정하는 방식으로 수정하였다.

후에 SRAM, FSM, Control Unit을 사용하여 키패드로 패턴을 입력하는 것을 구현하려 시도했지만 마감시간이 촉박하여 구현하지 못했다. 그리고 점수를 계산하는 것은 처음 계획에는 없었던 구현내용이지만 프로젝트를 진행하던 도중, 프로젝트의 완성도를 높이기 위해 타이머를 사용하여 추가로 구현하게 되었다.

## 5. 모듈 기능 요약

모듈명	기능
week11.v	모든 모듈을 통합하여 프로젝트를 실행
trigger.v	디바운싱을 방지함
LED.v	정답을 맞췄는지 확인함
Password.v	패턴과 입력값을 비교함
four_bit_reg_ce.v	4비트 이진수값을 레지스터에 저장함
PNU_CLK_DIV.v	각 모듈에 맞는 클럭값을 분주해서 할당함
simon.v	저장된 패턴값을 led에 표시함
d2b.v	키패드 입력값을 4비트 이진수로 변환함
count_8.v	세그먼트 디스플레이에 잔상효과를 주기 위해 사용함
eight_digit_shift_register.v	8자리 패턴을 각 레지스터에 저장함
score_calculator.v	시간에 따라 점수를 산출함
single_seg_decoder.v	싱글 세그먼트 디스플레이를 활성화함
timer.v	정답을 맞추는데 걸린 시간을 계산함
b2seg_bus.v.	이진수값을 a~g로 변환함
display_controller.v	array 세그먼트 디스플레이를 활성화함