

디지털논리로실험

[공지사항]

박 태 준

로봇공학과



일 정

주차	실험 및 강의 계획		
1	Verilog Lecture #1		
2	Verilog Lecture #2	FPGA Board & Quartus2 Introduction	
3	Verilog Lecture #3	Quartus2 Simulation, Lab 0 (Practice) Lab 1 : Switches, Lights, Multiplexers	
4	Verilog Lecture #4	Lab 2 : Characters and Displays	Report (Lab 1)
5	추석연휴		
6		Lab 3 : Numbers and Displays	Report (Lab 2)
7		Lab 4 : Adders	Report (Lab 3)
8	중간 시험		
9		Lab 5 : Latches, Flip-Flops, Registers	Report (Lab 4)
10		Lab 6 : Counters	Report (Lab 5)
11		Lab 7 : Clocks and Timers	Report (Lab 6)
12	(11/22)	황해 산업벨트 제조혁신 포럼 참석	
13	(11/29)	Lab 8 : Adders, Subtractors, Multipliers	Report (Lab 7)
14	대면 평가 (12/06)	Term Project 수행 #1	Report (Lab 8)
15		Term Project 수행 #2	
16	(12/20)	Term Project 발표	Report (Project)

대면 평가

- 장소 : 제5공학관 520호 (박태준 교수 연구실)
- 일시 : 12월 6일 (수) 3:00pm ~
 - 1인당 14분씩 평가
 - 평가순서

순번	1	2	3	4	5	6	7
시간	15:00	15:14	15:28	15:42	15:56	16:10	16:24
이름	신혜영	조민수	유동하	김준범	유승혁	정희종	최정훈

순번	8	9	10	11	12	13	14
시간	16:38	16:52	17:06	17:20	17:34	17:48	18:02
이름	박현우	유남기	송용준	황병준	정현철	임지수	이동환

- 범위 : Lab 1 ~ 8

Term Project

- 팀 편성

- 4인 1팀을 원칙으로 하나, 1~3인 팀도 허용함
- **11/29 (수) 6:00pm**까지 팀 편성 완료

- 실험장비

- 실험조교 허락 하에 실험실 내에서 사용

- 최종발표 : **12/20 (수) 3:00pm ~**

- 최종보고서, 발표자료 및 Verilog Code를 **당일 오전 9시까지** 조교에게 이메일 제출
- 팀당 **30분 발표 및 시연**, 15분 질의 응답으로 진행
- 실제 동작을 장비를 활용해 시연해야 함

Term Project

- 과제 구성

- 필수 과제 : 데이터 오류 검출/정정 모듈 설계 100%
- 선택 과제 1 : 블랙잭 25% (Extra Credit)
- 선택 과제 2 : 계산기 25% (Extra Credit)

- 평가 기준

- 팀간 상대평가이며, 팀원간 모두 같은 점수 부여

- 평가 지표

- 과제별 세부 목표 달성도 50%
- 발표 20%
- 창의성 15%
- 보고서 15%

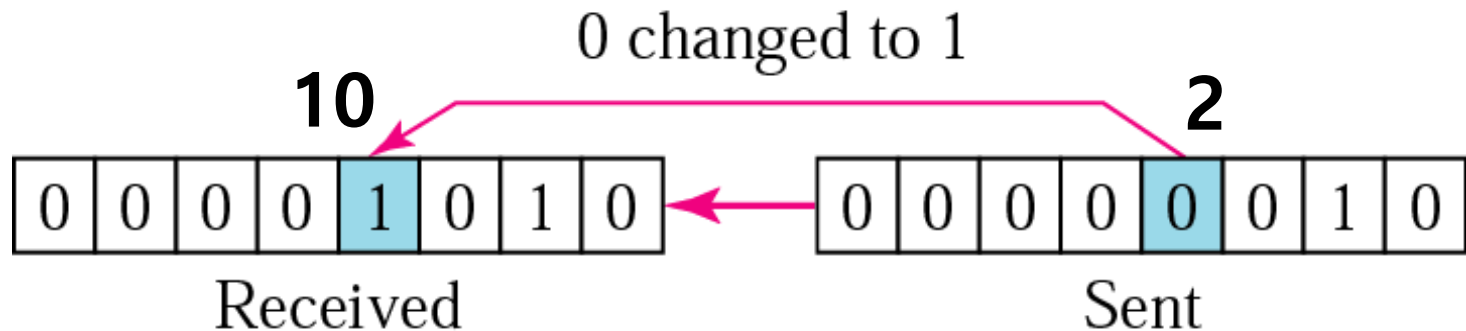
필수과제

데이터 오류 검출/정정 모듈 설계

데이터 오류 검출/정정 모듈 설계

- 개요

- 데이터 전송 시 회선 상의 오류 또는 해킹에 의해 송신자가 보낸 정보가 아닌 다른 정보로 바뀌어 수신되는 상황이 발생함
- 따라서 오류를 찾아내고 정정하는 방법이 필요함



데이터 오류 예시

데이터 오류 검출/정정 모듈 설계

- **목표**

- 송/수신간에 데이터 오류를 자동으로 검출하고 정정하는 모듈을 설계

- **세부목표**

- 패리티 검사 (Parity check) 및 응용 설계
 - 1. 단순 패리티 검사
 - 2. 패리티 검사 및 오류 정정
 - 3. 2bit 에러 검사

데이터 오류 검출/정정 모듈 설계

- 패리티 검사

- 데이터에 패리티 비트를 붙여서 1의 전체 개수가 짝수 혹은 홀수가 되도록 한다.
- 예제 (송신자)
 - 송신자가 'world'라는 단어를 보내고자 한다. ASCII를 사용하면 다섯 글자는 다음처럼 코드화 된다.

1110111 1101111 1110010 1101100 1100100

- 다음은 실제 전송되는 비트를 보여주고 있다.

11101110 11011110 11100100 11011000 11001001

데이터 오류 검출/정정 모듈 설계

- 패리티 검사

- 데이터에 패리티 비트를 붙여서 1의 전체 개수가 짝수 혹은 홀수가 되도록 한다.
- 예제 (수신자)
 - 'world'라는 단어가 전송 중 변환되었고, 이를 수신자가 수신했다고 가정하자.

11111110 11011110 11101100 11011000 1001001

- 수신자는 각 글자에서 1의 수를 세고 짝수와 홀수(7, 6, 5, 4, 4)임을 알아낸다. 수신자는 데이터가 변환되었음을 알고 그 단어를 버리고 재전송을 요청한다.

데이터 오류 검출/정정 모듈 설계

- **설계 1** (단순 패리티 검사)

- SW를 이용하여 16 진수 4자리 숫자를 입력 받아 HEX 3-0에 표시한다.
- KEY를 누르면 4자리 숫자 중 랜덤으로 **1비트 오류**를 발생시키고 오류가 난 결과를 HEX3-0에 표시한다.
- 패리티 검사 모듈을 설계하고 오류 데이터를 검사하여 HEX4에 1의 자리가 에러이면 0, 10의 자리가 에러이면 1, 100의 자리가 에러는 2, 1000의 자리가 에러는 3을 표시한다.

- **설계 2** (패리티 검사 및 오류 정정)

- 설계1의 상황을 가정한다.
- 오류가 발생한 비트를 찾아서 정정하는 모듈을 설계하라.
- KEY를 누르면 정정된 결과를 HEX3-0에 표시한다.

데이터 오류 검출/정정 모듈 설계

- **설계3** (2bit 오류검사)

- 다음처럼 2비트 오류가 나는 상황을 가정하자.

- 예시1: 1010101 1111000 -> 1010101 1011001

- 예시2: 1010101 1111000 -> 1110101 1011001

- 위와 같은 상황에서 오류 검출을 위한 검사 모듈을 만들어라.

****제시된 조건 중 동작 시나리오, 입력방법, 출력방법 등은 자유롭게 변경 가능하며, 추가기능 또한 자유롭게 추가 가능함**

****각 설계간 핵심 기능인 패리티 오류검사 모듈, 정정 모듈, 다중 비트 오류검사 모듈만 따로 실행 후 Total logic elements, fmax 값을 보여줄 것**

선택과제 1

블랙잭

블랙잭

- 개요

- 블랙잭은 세계의 카지노에서 가장 널리 행해지는 플레이카드 게임이다.
- 이 게임은 21을 넘지 않는 한도 내에서 딜러와 겨루어 숫자가 높으면 이기는 게임이다.

- 상세규칙

- 처음 시작 시 카드 2장을 받음
- 먼저 받은 카드 두 장의 합이 21에 못 미치면 히트(Hit)라고 말한 뒤 한 장을 더 받을 수 있고 멈추려면 스탠드(Stand)라고 말한다.
- 카드의 합이 16이하 일 경우 무조건 한 장을 더 받아야 하고 17이상의 경우 멈출 수 있다.
- 카드의 합이 딜러보다 먼저 21이 되거나 21에 가깝게 되면 이기고, 카드를 받았는데 21을 초과하면 지게 된다.
- 에이스 카드는 1이나 11로 취급할 수 있고, 10,J,Q,K는 모두 10으로 계산한다.

블랙잭

- 설계조건
 - 블랙잭 규칙을 기반으로 자유롭게 설계한다.
 - 예 : 플레이어 수, 제한 시간, 게임머니, 게임이 연속적으로 이어지는 상황 등을 고려할 수 있다.

선택과제2

계산기

계산기

- 개요

- 사칙연산이 가능한 계산기를 자유롭게 설계한다.
- 기본 기능으로는 "+, -, ×, ÷" 이며 음수를 지원한다.
- 반드시 논리회로를 이용하여 설계 한다.
 - +, -, *, / 등의 연산기호를 이용하여 구현하지 않음
 - 이를 어길 시 0점 처리
- 이외에 square, factorial, log, exp 등의 추가 연산자는 원하는 데로 추가 할 수 있다.