

2022년도 1학기 논리회로 설계 프로젝트

C078-2, C079-3

✓ 설계 프로젝트 내용

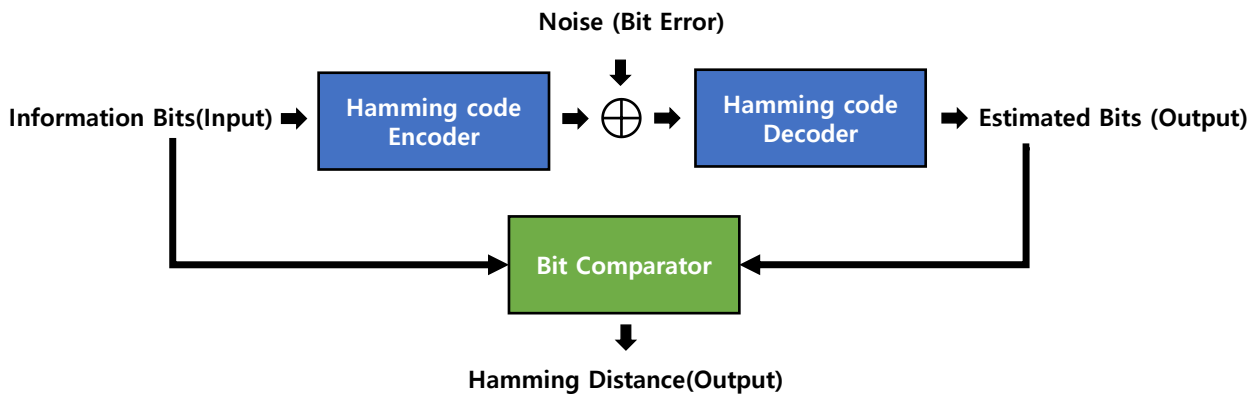
□ (17, 12) Hamming code Design

■ Input: Information bit (12-bit)

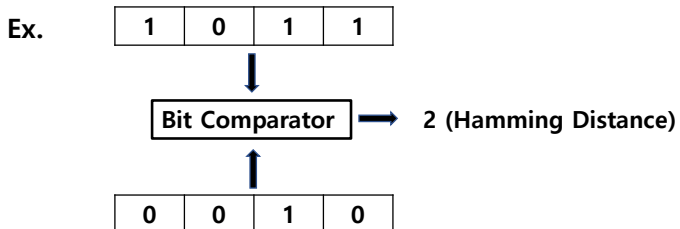
■ Output: Estimated bit (12-bit), Hamming distance (4-bit)

- 1) Information bit를 입력 받은 Hamming Encoder는 17-bit Codeword 출력
- 2) Hamming Decoder로 송신하는 과정에서 Noise가 더해져 1-bit Error 발생
- 3) 2)에서 Noise가 추가된 17-bit Codeword를 입력 받은 Hamming Decoder는 오류를 정정한 후 Estimation bit를 출력
- 4) 3)에서 출력된 Estimation bit와 기존 Information bit를 입력 받은 Comparator는 Hamming distance를 출력
- 5) 결과적으로 12-bit 정보를 입력 받아 송, 수신하는 시스템을 Verilog HDL로 설계

● Schematic



● Hamming distance



◆ Binary Number는 “MSB, ..., LSB” 로 통일 (Unsigned Binary Number System)

◆ (17, 12) Hamming Code

■ 12-bit의 Information Bits + 5-bit의 Parity Bits → 17-bit의 Hamming Codeword

Verilog HDL 설계 내용

A. (17, 12) Hamming Encoder 설계

1) Information bit (12-bit)를 수신하여 17-bit Codeword 생성하는 Module인 “ham_enc.v”를 설계

B. (17, 12) Hamming Decoder 설계

1) A에서 생성한 Codeword를 수신하는 과정에서 1-Bit Error 발생

2) Error가 발생한 Codeword를 Error-Correcting하여 Estimated bit를 출력하는 Module인 “ham_dec.v”를 설계

C. Bit-Comparator 설계

1) Information bit (12-Bit)와 Estimated bit (12-Bit)를 입력 받아 두 열의 Hamming Distance (4-Bit)를 출력하는 Module인 “bit_com.v”를 설계

D. Top Module 설계

1) “top.v” 에 Verilog 강의노트를 참고하여, Schematic에 맞게 A, B, C의 Module Instantiation, Wire 연결 등 Structural Description 방식으로 최상위 모듈 설계

E. Test Bench 설계

1) A, B, C에서 설계한 총 3개 모듈들에 대해서 배포한 Test Bench 파일에 Module 당 5가지 이상의 Test Case를 작성하여 설계한 Sub-Module들이 정상적으로 동작하는지 확인

2) “tb_top.v” 에 배포한 Test Bench 파일 및 Verilog 강의노트를 참고하여, Top Module을 Instantiation한 후 5가지 이상의 Test Case를 작성하여 자신이 설계한 Top Module이 정상적으로 동작하는지 확인

◆ 기본적으로 입출력 인터페이스 등 어느 정도 틀이 짜인 Verilog HDL Code 제공

■ 사전 설계된 코드 내용 변경 금지, 이외에는 문제의 조건에 맞는 선에서 자유롭게 코딩

◆ 모든 코드에 주석을 이용한 간략한 설명 명시 (Code 설명 등)

■ ModelSim은 한글 지원이 불가해 영문으로 작성, 또는 보고서에서 따로 주석 가능

◆ 모든 Verilog HDL 파일 중 하나라도 컴파일 에러가 발생하면 실격 처리

◆ 강의노트 및 조교가 제공한 코드 외, 외부 코드 Copy 시 실격 처리

✓ 보고서 내용

1. 공학 인증 표지
2. 문제 설명 (1 page 내로 간략히)
3. A, B, C, D에서 설계한 자신의 Code 분석 (자신이 이렇게 설계한 원리, 구조, 동작 등 자세히)
4. 모든 Sub-module들의 Simulation 결과 캡처 및 분석
5. 자신이 설계한 Top Module의 Test Bench Code 분석, Simulation 결과 캡처 및 분석
6. 기능별 달성도 평가 (10점 만점 기준)

◆ 보고서에 코드 첨부 시, 가시성을 위해 하이라이팅하여 첨부

- hilite.me 접속 → 'Source Code' 란에 코드 복사/붙여넣기 → 'Language' Verilog로 설정 → 'Highlight!' 클릭 → 하단 'Preview' 란의 하이라이팅된 코드를 복사하여 첨부 (다른 사이트/툴 이용 가능)

◆ 너무 간단하거나, 너무 방대하지 않게 자율적으로 분량 조절

◆ 시뮬레이션 분석을 용이하게 하기 위해 Wave의 Radix 변경 가능

✓ 제출 시 유의 사항

- 제출물 : 보고서 (한글 파일 또는 PDF 파일), 모든 Verilog HDL Codes (.v 파일)

하나의 압축 파일로 압축

파일 이름 : "학번_이름_반.zip" Ex) "202124109_이용제_오전반.zip"

- 제출 기한 : 6월 1일 수요일 자정 (오후 11시 59분)

- 제출 방식 : Bb 과제 제출

◆ 불이익을 당하지 않게, 제출 전후로 모든 파일 검토 및 확인