Lab 1. 논리회로 기초

2024. 03. 20.  
디지털 시스템 설계 (CSED273)

1. 개요

기초 논리 게이트로 회로를 구성하고, Verilog로 구현한다. 세부적인 학습 목표는 다음과 같다.

* Xilinx Vivado HDL 통합 개발 환경 적응
* Verilog 문법 습득
* 회로의 작동을 확인하기 위한 Testbench 작성 및 시뮬레이션 사용 방법 습득
* Functionally complete 개념 이해

2. 이론적 배경

1) Positive Logic / Negative Logic

논리식을 실제 전자 회로로 구현할 때 불 대수의 참과 거짓을 High (높은 전압)와 Low (낮은 전압)에 임의로 대응할 수 있다. 예를 들어 Positive logic (Active high)의 경우 참을 High, 거짓을 Low로 표현하고, Negative logic (Active-low)의 경우 참을 Low, 거짓을 High로 표현한다.

이 둘은 논리 회로를 전기적으로 표현하는 방법만 다를 뿐 기능적인 차이는 없다. 단, 실제 부품을 사용하여 회로를 구성할 때는 Active-high와 Active-low 부품이 섞여 있을 수 있으므로 주의해야 한다.

|  |  |  |
| --- | --- | --- |
| 전기 신호 (예시) | Active high에서의 값 | Active low에서의 값 |
| **0V (Low)** | False | **True** |
| **5V (High)** | **True** | False |

2) HDL (Hardware Description Language)

Hardware Description Language는 디지털 시스템의 논리 회로 구조를 표현하기 위한 언어다. 주로 사용되는 언어로는 Verilog와 VHDL이 있으며, 이번 디지털 시스템 설계 강의에서는 Verilog를 사용한다. SystemVerilog와는 다른 언어로 지원하는 기능에 차이가 있으므로 주의가 필요하다.

3) Functionally complete set

어떤 논리 연산의 집합으로 모든 불 대수식을 표현할 수 있을 때 이 집합은 Functionally complete이다. 예를 들어, 기초 연산만으로 이루어진 {AND, OR, NOT}은 Functionally complete이다. 따라서 어떠한 집합이 Functionally complete임을 증명할 때, 포함된 연산을 조합하여 AND, OR, 그리고 NOT 연산을 만들어 낼 수 있음을 보이면 된다.

3. 실험 준비

1) 오리엔테이션에서 다룬 Vivado 사용법, Verilog 문법, 그리고 Testbench 작성 방법을 숙지한다.

2) NOT, OR, AND, NOR, 그리고 NAND 총 다섯 개 연산의 진리표를 작성한다.

4. 실험

0) 공통 유의사항

* 주어진 코드에서 /\* Add your code here \*/ 로 주석 처리된 부분만 수정하여 구현한다.
* 각 모듈은 Gate-Level Modeling으로 구현한다.

1) AND 게이트 구현 – lab1\_1.v, lab1\_1\_tb.v

ㄱ. AND 게이트를 Verilog로 구현한다.

ㄴ. Testbench를 작성하고 시뮬레이션을 수행한다.

* 총 10ns동안 시뮬레이션을 수행한다.
* AND 게이트의 두 입력을 각각 1㎱와 2㎱마다 한 번씩 반전시킨다.
* 0㎱일 때 입력의 초기값은 모두 1로 설정한다.

ㄷ. 시뮬레이션 결과를 확인한다.

**\* 보고서 필수 내용**

* 전체 시뮬레이션 파형 캡쳐

2) Functionally complete 집합 구현

ㄱ. 주어진 연산만으로 코드에 제시된 모듈을 구현하여 {AND, OR, NOT} 집합을 완성한다.

1. {OR, NOT} - lab1\_2\_i.v
2. {AND, NOT} - lab1\_2\_ii.v
3. {NAND} - lab1\_2\_iii.v
4. {NOR} - lab1\_2\_iv.v

ㄴ. RTL Analysis -> Elaborated Design -> Schematic 기능으로 작성한 회로를 확인한다.

**\* 보고서 필수 내용**

* Schematic 기능으로 생성한 회로도 캡처

▸ 실험 iii.과 iv.는 하위 모듈의 회로도도 캡처해야 한다. (하위 모듈은 “+” 버튼을 눌러 확인 가능, 아래 그림 1 참고)

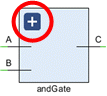


Figure 1.

5. 제출

{학번}\_lab1.zip으로 다음 파일을 압축하여 PLMS에 제출한다.

* lab1\_1.v // 코드
* lab1\_1\_tb.v // Testbench
* lab1\_2\_i.v
* lab1\_2\_ii.v
* lab1\_2\_iii.v
* lab1\_2\_iv.v
* lab1\_report.pdf // 보고서