

Logic Design Lab Report: Week 10

2013-11826 임주경

1. Introduction

Two-Digit-Counter 를 구현해서 0~99까지 count 하도록 한다. Toggle 과 Reset Input을 포함시킨다. 구현한 Counter 를 Logic Design Board 에 implement 한다.

2. Implementation

구현은 Frequency Divider, Counter, BCD to 7 Segment를 이용해 구현한다. 구현 방식은 Behavioral description 사용하였다. Frequency Divider를 구현한 freqdiv.v 의 코드는 always 구문과 if 구문을 사용하였으며, 보드의 50MHz 클락을 1Hz 클락으로 바꿔준다. Reset의 기능은 Counter의 Reset과 겹치게 되어 불필요하다고 생각되어서 코드에서 제외하였다. Counter를 구현한 b7counter.v 의 코드는 인풋으로 클락, 리셋, 토글을 받는다. 클락의 positive edge 마다 리셋과 토글의 상태를 if 구문을 통해 확인하며, 확인한 상태에 맞게 현재 count를 출력하고, 다음 count가 진행된다. 리셋이 눌리면 count를 0으로 초기화 하고 출력 결과도 0으로 결정한다. 토글은 스위치가 눌리면, reverse로 이름 붙인 register 값이 변경된다. 0일때는 증가 방향, 1일때는 감소 방향으로 count 한다. BCD to 7 Segment로 Output을 출력하기 위해서, 10의 자리와 1의 자리 숫자는 각각 다른 Output으로 지정한다. BCD to 7 Segment는 이전의 실습에서 구현했던 bcd_7decoder.v 를 사용하였다. 마지막으로, Two-Digit-Counter를 구현한 TDC.v 의 코드에서 인풋 클락을 Frequency Divider에 입력하면, 출력되는 아웃풋 클락을 Counter의 인풋 클락으로 연결해주고, Counter의 아웃풋 결과로 나온 10의 자리 숫자와 1의 자리 숫자를 BCD to 7 Segment 2개의 각각 인풋으로 지정해주었다.

Logic Design Board에 mapping은 리셋은 P47의 tactile switch, 토글은 P54의 tactile switch, 클락은 P57의 50MHZ의 Oscillator에 mapping 하였고, 나머지 아웃풋은 7 Segments LEDs에 알맞게 mapping 하였다. Mapping의 내용은 코드 TDC.ucf 에 명시 되어있다.

3. Result

구현한 Logic Design Board의 정상적인 작동 영상을 같이 압축하여 첨부하였다. 구현 결과는 모두 정상적으로 작동한 것을 확인할 수 있다. 1초마다 카운트가 발생하며, 99다음은 0으로 반대 방향으로 진행할 때는 0다음은 99로, 그 외에 리셋 기능도 정상적으로 작동한다.

4. Conclusion/Discussion

결과는 이상없이 출력됨을 확인할 수 있었다. 다만, Counter 초기 구현 과정에서 Reset이 Frequency Divider의 Reset과 겹쳐서 정상적으로 작동하지 않음을 확인할 수 있었는데, 이는 Reset이 1이 되었을 때, Frequency Divider의 아웃풋 클락이 0이 되고, Counter의 always @(posedge 인풋클락)과 엇갈려서, Reset 알고리즘이 정상적으로 작동하지 않는 것이 문제임을 알 수 있었다. 위와 같은 문제를 해결하기 위해서, Frequency Divider의 Reset을 없애고 구현하였다. 그 결과, 이상 없이 잘 작동할 수 있었다.