8주차 결과보고서

전공 : 아트앤테크놀로지 학년 : 3학년 학번 : 20191098 이름 : 백승주

1 4 비트 크기의 이진수 숫자를 입력하면 7-segment display로 16진수 숫자를 0부터 F까지의 숫자를 나타내는 소스 코드를 작성한다. 이 소스 코드를 작성하기 위해 진리표와 카르노 맵을 작성해 output에 대한 식을 작성한다. 또한 작성한 코드를 testbench 코드를 통해 시뮬레이션 결과를 확인하고 소스코드에 대한 결과를 fpga 보드를 통해 확인한다.

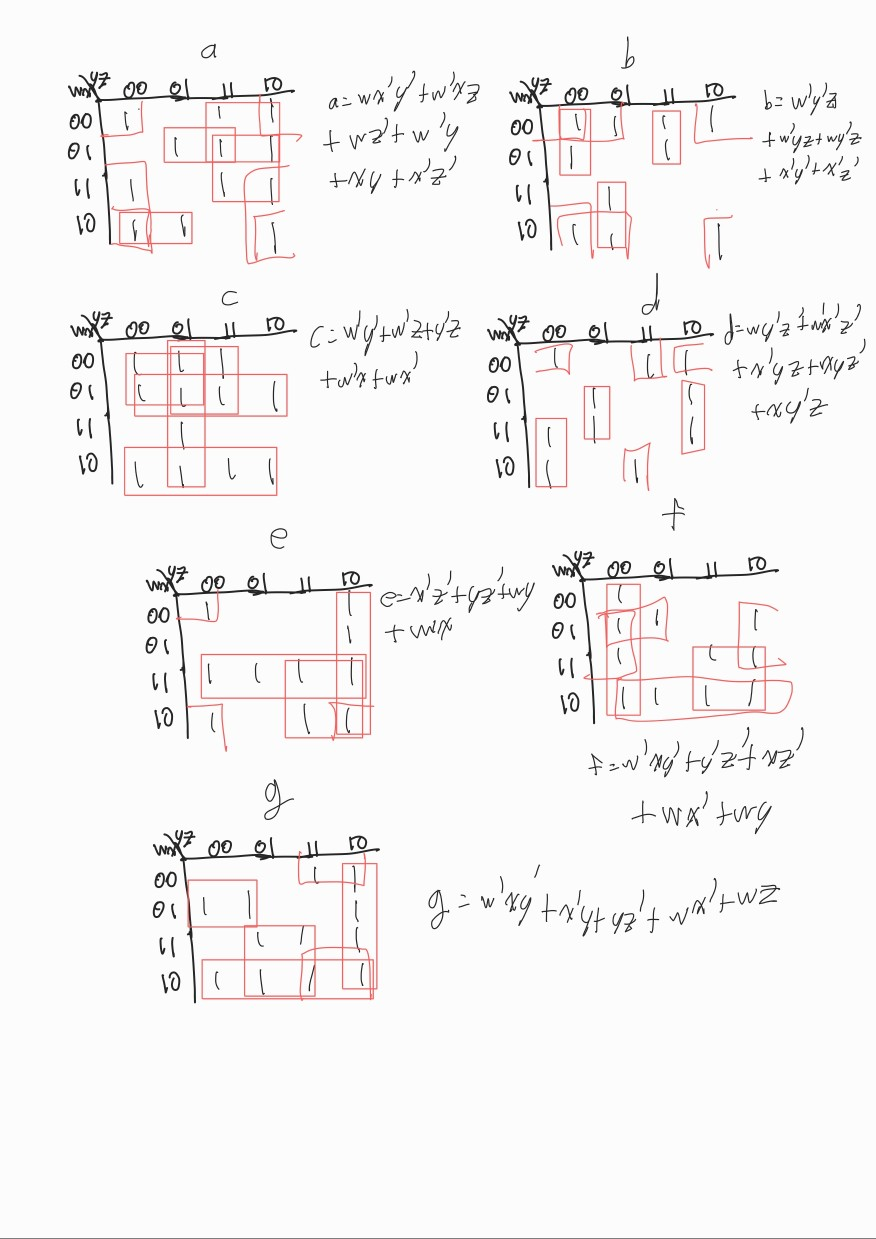
2.



위 그림을 토대로 input w,x,y,z, 에 대해 다음과 같은 진리표를 얻을 수 있다.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **W** | **X** | **Y** | **Z** | **a** | **b** | **c** | **d** | **e** | **f** | **g** |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |

위 진리표를 토대로 다음과 같은 카르노 맵과 식을 얻을 수 있다.

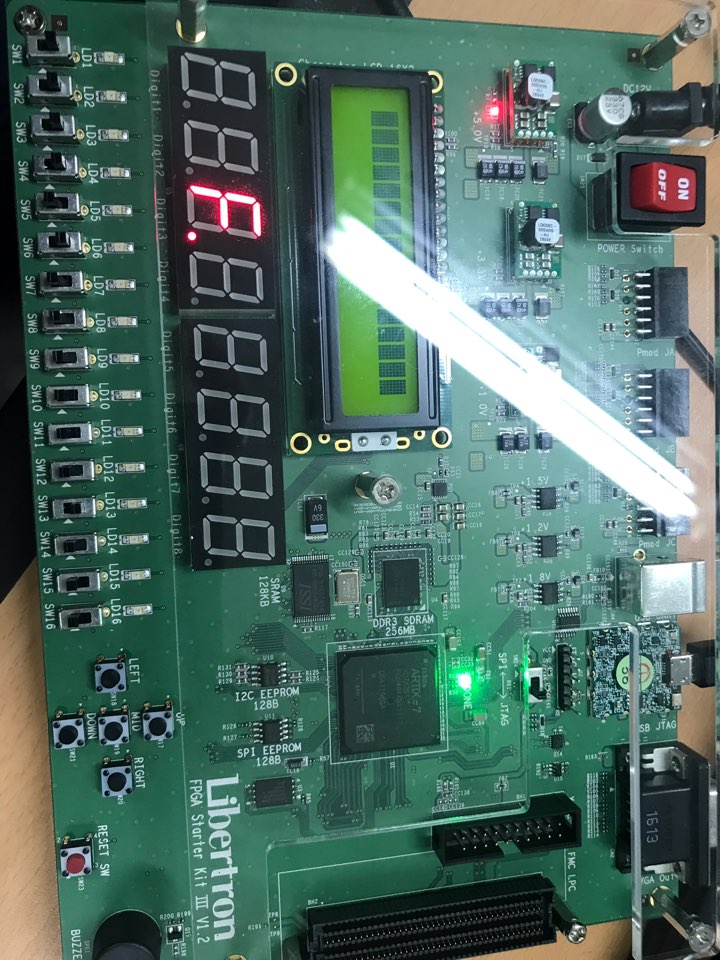


위의 카르노 맵에서 얻은 식을 토대로 다음과 같은 Verilog 코드를 작성할 수 있다.

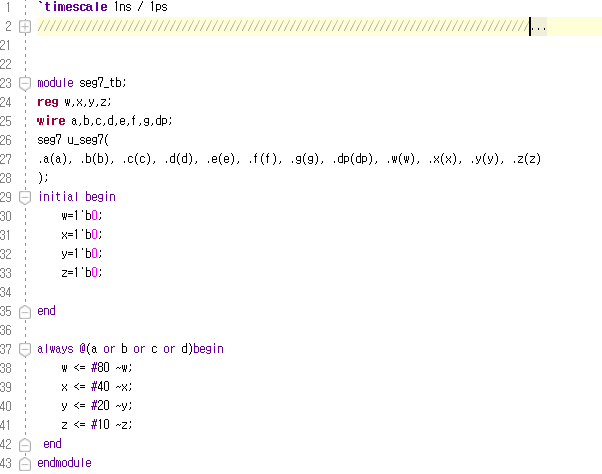
텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

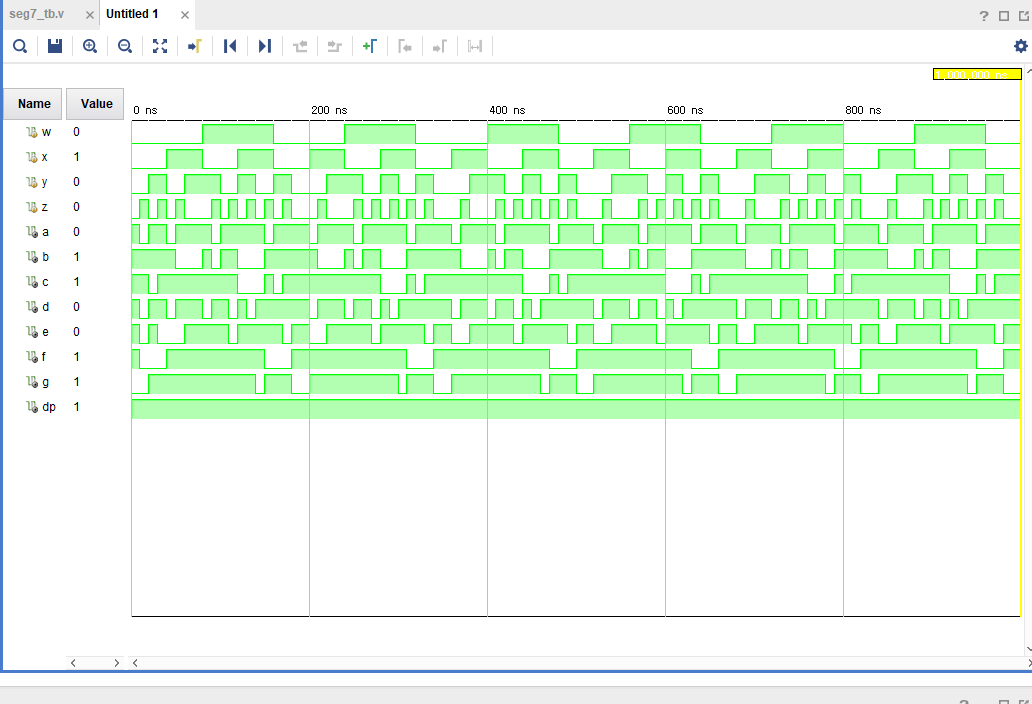
여기서는 기존의 카르노 맵에서 구하지 않았던 dp와 outd라는 input이 추가된 걸 알 수 있는데 먼저 에는 숫자의 점을 나타낸다. 다음의 사진을 볼 때 F 옆에 나타나는 점이 바로 dp이다.



이 점은 0~15 숫자 모든 경우에서 다 나타나기 때문에 1로 설정해준다. 또한 outd 이 값은 digit을 선정하기 위해 설정한 값이다. Fpga의 8개의 digit 중 한 곳을 활성화하기 위해 만든 값으로 모든 input들에 대한 or 값으로 어떠한 input 하나만 1이 되어도 digit은 활성화되야하기 때문에 위와 같이 설정했다. 다음의 코드들은 위 Verilog 코드에 대해 testbench 소스 코드를 작성한 결과이다.



위의 testbench 파일을 통해 확인한 시뮬레이션 결과는 다음과 같다.



3. 위 시뮬레이션 결과를 통해서 4개의 input을 이용해 4-bit 값의 숫자를 입력할때마다 input a,b,c,d,e,f,g, 가 어떤 값을 가지는지 확인할 수 있었다. 예를 들어 위 시뮬레이션 화면의 예처럼 w=0,x=1,y=0,z=0 일 경우 이 input들이 나타내는 값이 100, 4이기 때문에 7-segment에서 4를 나타내기 위해 해당되는 파트들인 b,c,f,g,에 부분의 값이 1이 되었다. 즉 0~15까지의 모든 수들이 fpga 보드에 16진수 값으로 해당 값이 표현되는 것을 확인할 수 있었다.

4. 7-segment 코드를 설계할 때 처음에는 outd라는 변수를 설정해 digit 3 에 이 변수를 연결하지 않았고 fpga 보드에는 아무것도 출력되지 않았다. 이 digit의 역할에 대해 알아보자. Digit에 연결한 outd의 값은 7개의 입력 a, b, c, d, e, f, g, dp의 논리합이었는데 이 값이 1이면 7-segment display에 출력될 값이 존재한다는 뜻이고 0이면 출력값이 없다는 뜻이다.

Fpga 보드는 7-segment display를 제어하기 위한 출력 핀이 필요하다. 7-segment display 디코더 모듈은 a, b, c, d, e, f, g, dp와 같은 입력 핀을 사용해서 출력할 숫자나 문자 등의 값을 비트 패턴으로 변환하고 변환된 비트 패턴은 digit으로 전달되고 이러한 digit을 7-segment display 출력 핀에 전달하여 실제로 디스플레이를 제어하게 된다 즉 digit은 7-segment display에 출력될 값을 정하는 비트 패턴이다.