6주차 결과보고서

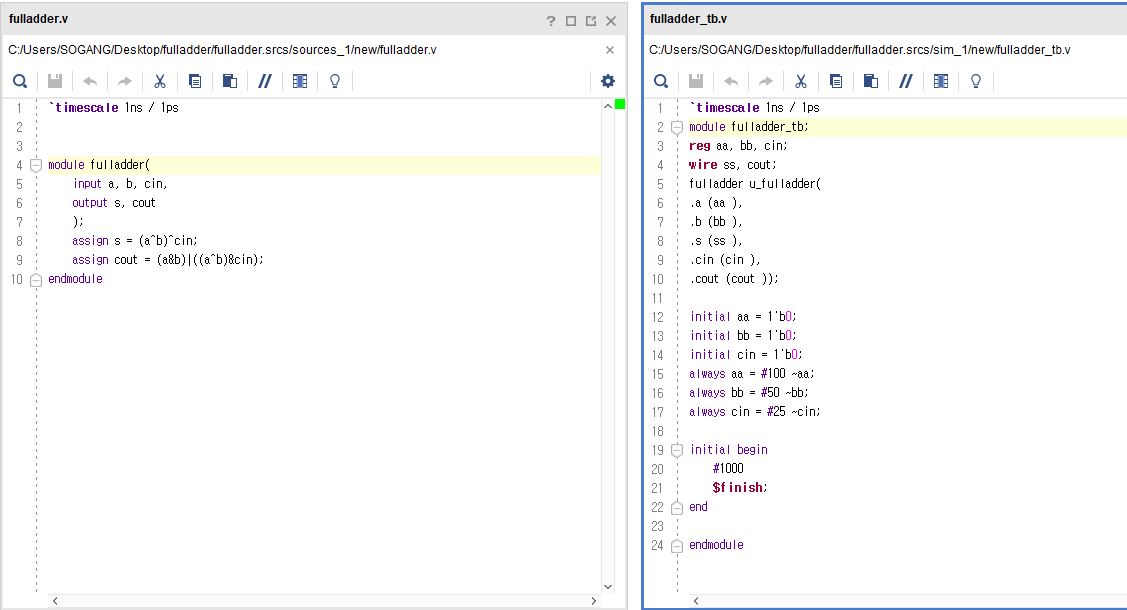
전공: 컴퓨터공학과 학년: 3학년 학번: 20191612 이름: 윤기웅

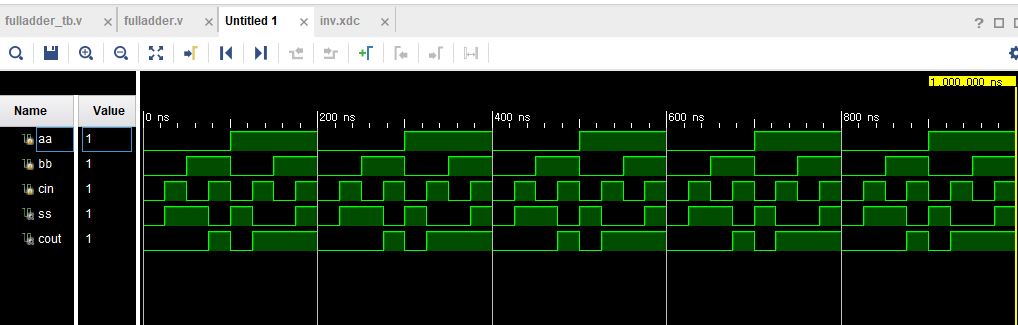
1. 가산기와 감산기의 개념을 이해하고 부호 변환기를 이해한다. 이후 verilog를 사용해서 다양한 가산기, 감산기, 변환기를 구현하고 회로의 동작을 직접 확인 해본다.

2.

1) full adder : Half Adder는 이전의 자릿수의 덧셈 결과를 반영하지 않지만 full adder는 이전 자릿수의 덧셈 결과를 반영해서 결과값을 만들어낸다. 그래서 s는 sum을 의미하고 3개의 입력값 중에서 홀수의 개수가 1일 때 1이 된다. 즉, 1개만 1이거나 3개가 모두 1인 경우에 1이 된다. 그리고 cout은 캐리를 의미한다. (a&b)|((a^b)&cin)의 값을 갖는다.

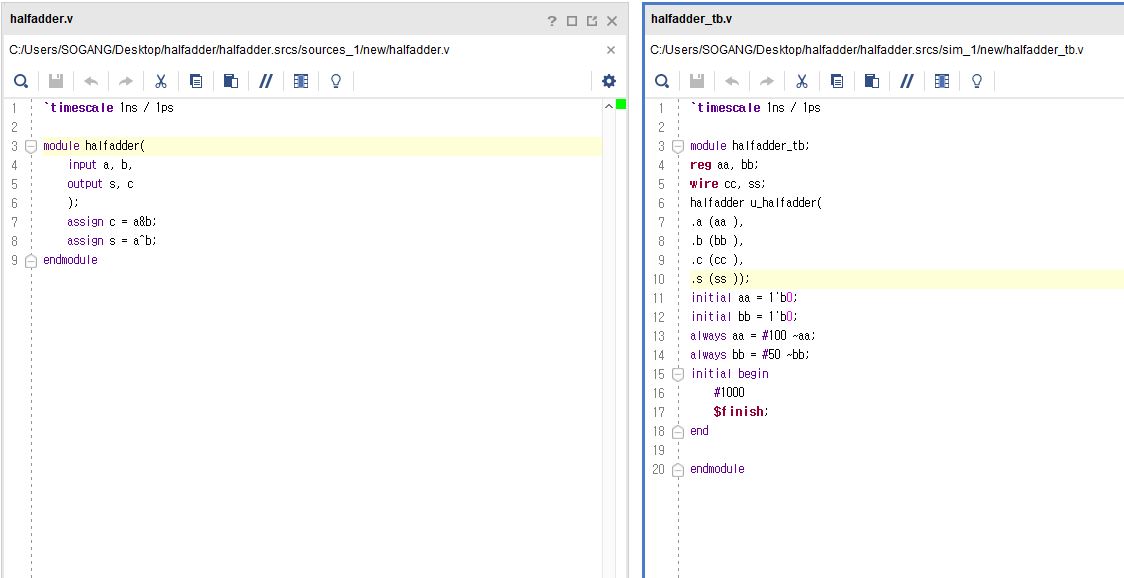
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b |  | S(Sum) |  |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

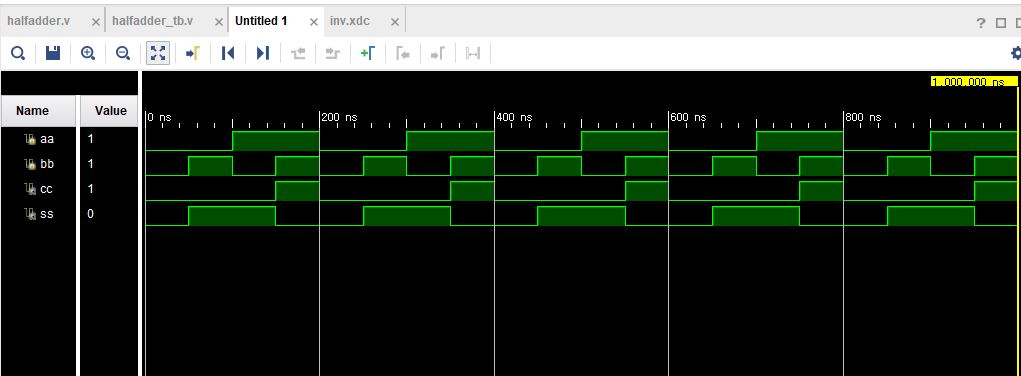




2)half adder : 입력값으로 a,b만 존재하고 이전 자릿수의 계산 결과는 반영하지 못한다. 결과값 중에서 c는 캐리를 의미하므로 입력값이 모두 1인경우 캐리가 1이 된다. 그리고 결과값 중 하나인 s는 그 자리수의 결과를 받는다. 그래서 입력값 중에서 하나만 1일 때 s가 1이 된다.

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | s(sum) | c(carry) |
| 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 |



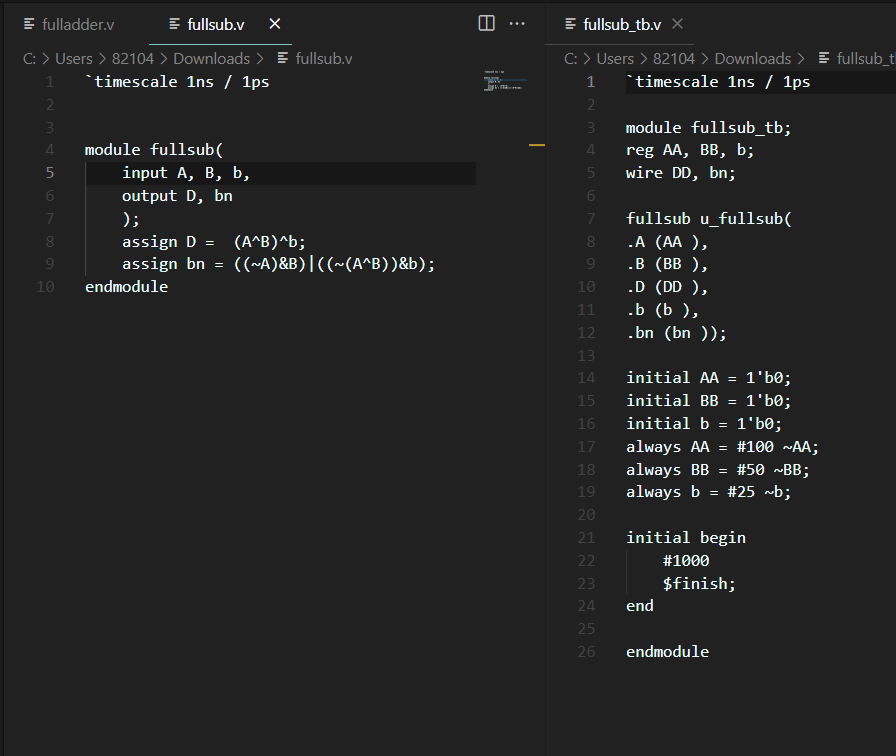


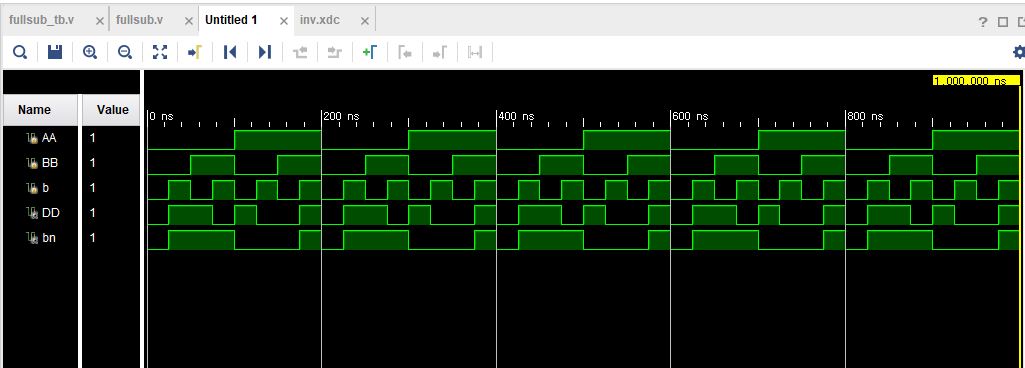
3.

1)full substracter

이전 자릿수의 뺄셈 연산의 결과를 반영할 수 있는 것이다. 입력으로 A,B,b가 들어오고 결과값으로 D와 bn이 주어진다. D는 입력값 3개의 xor 연산을 한 값이다. 그래서 1개만 1인 경우와 3개 모두 1인 경우에 D가 1이 된다. bn은 ((~A)&B)|((~(A^B))&b)의 연산 결과로 정해진다. 높은 자리의 수를 빌려와서 연산을 해야 하는 경우에 bn은 1이 된다.

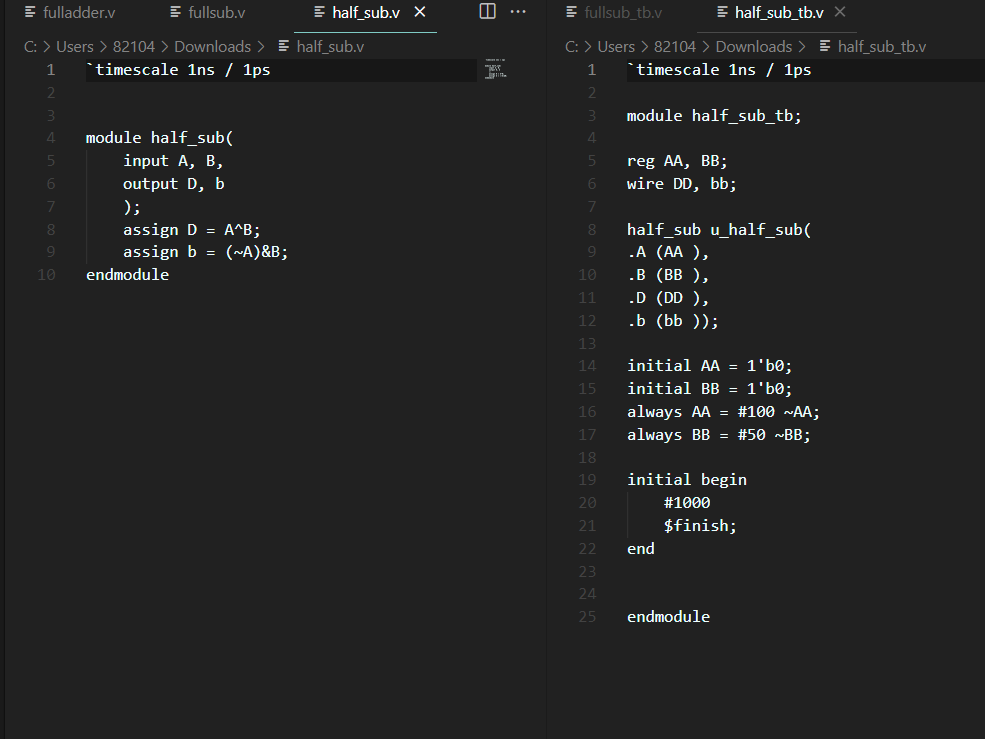
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | b | D | bn |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

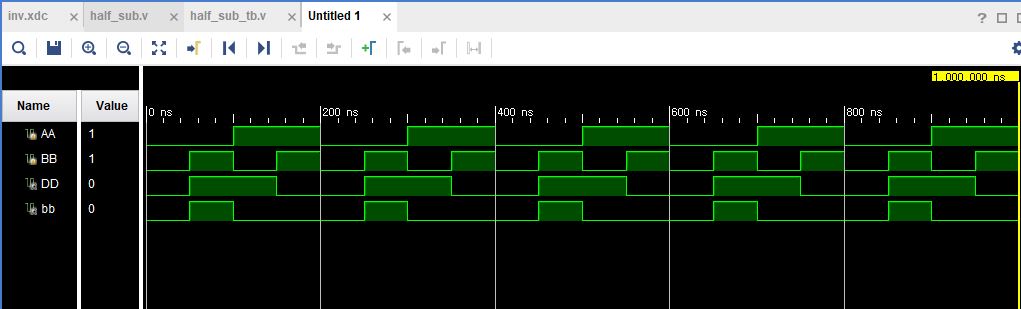




2)half substracter : 이전 자릿수의 뺄셈 결과를 반영하지 못한다. 입력값은 A, B로 두 개이고 결과값은 두 개이다. D,b가 결과값이고 b는 앞의 자리에서 뺄셈을 위해서 1을 빌려온 경우에 1이 된다. D는 뺄셈을 수행한 그 자리의 수이고 입력값 둘 중 하나만 1인 경우에 1이 된다. b는 ~A와 B의 and 연산을 한 값이다. 그래서 A가 0이고 B가 1이 되는 경우에만 b가 1이 된다.

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | D(Difference) | b(Borrow) |
| 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 |

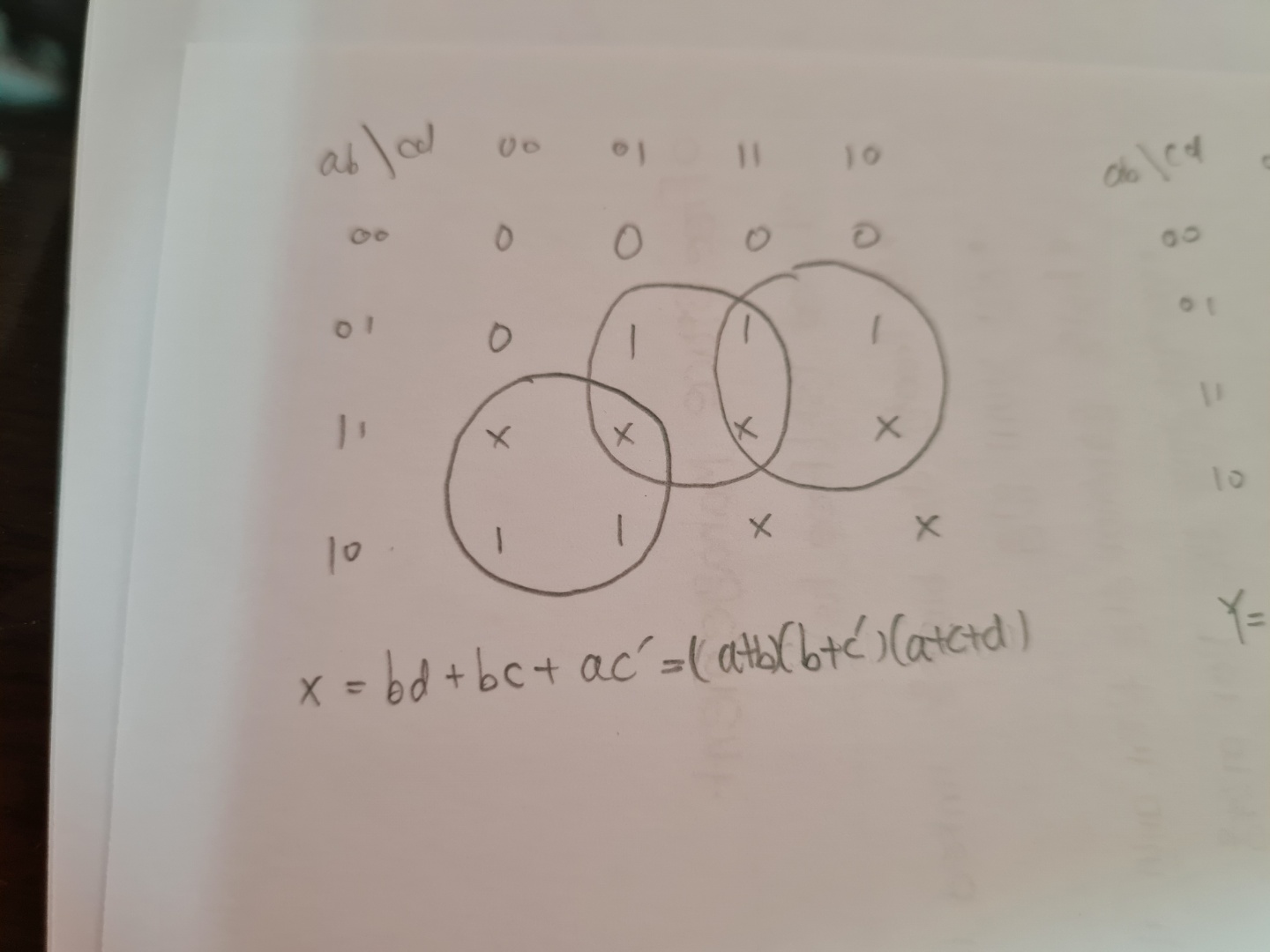


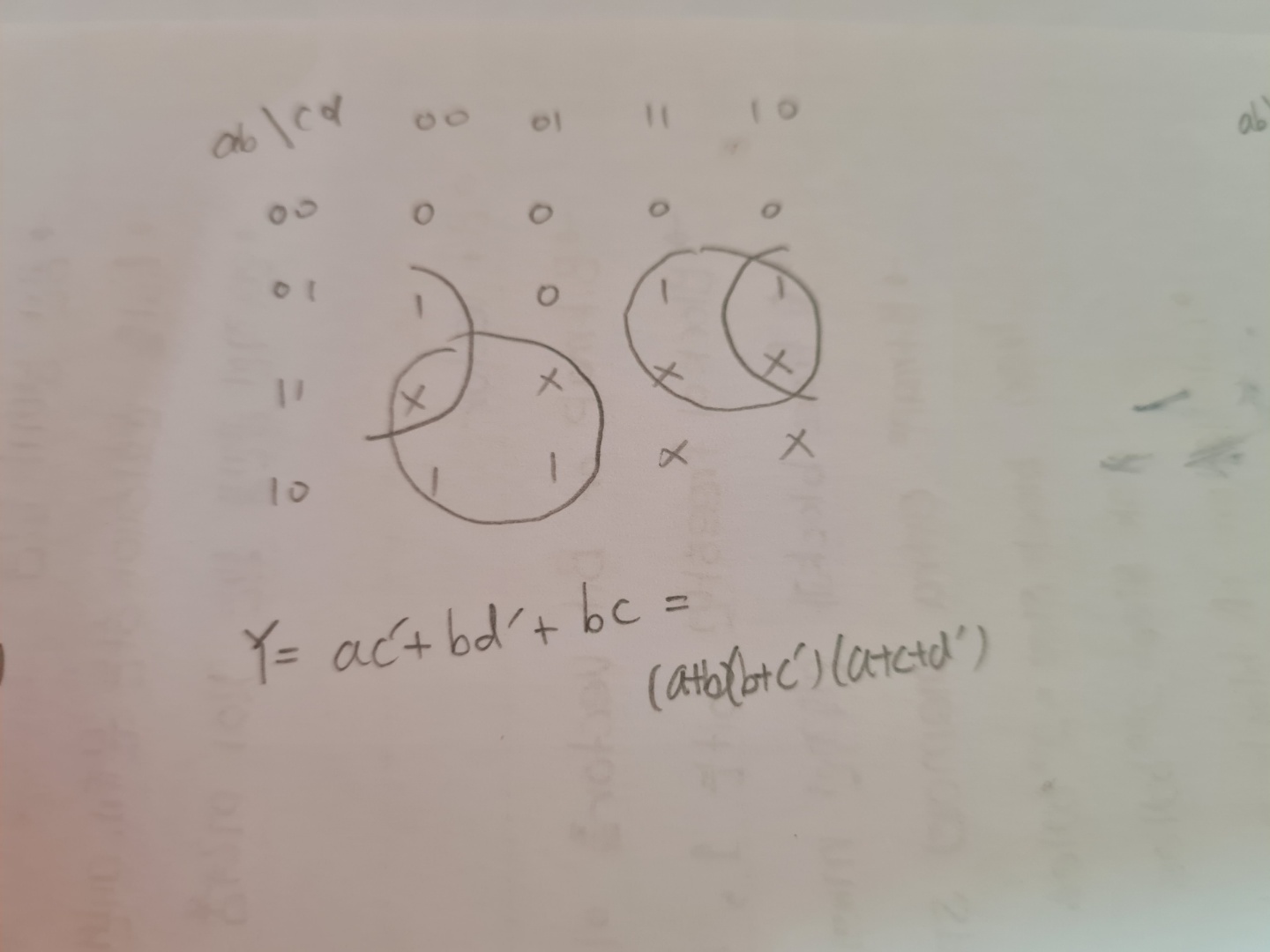


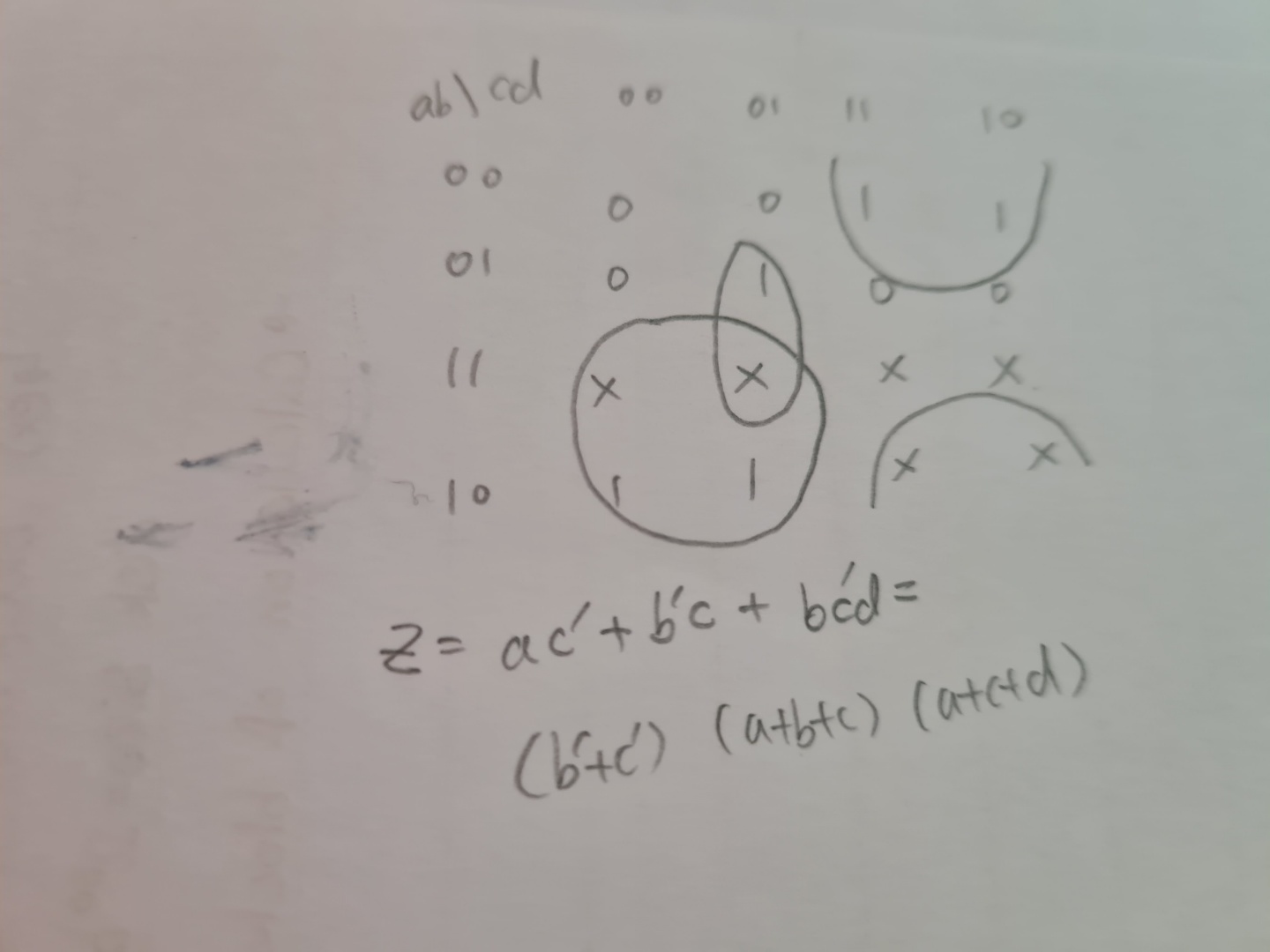
4.

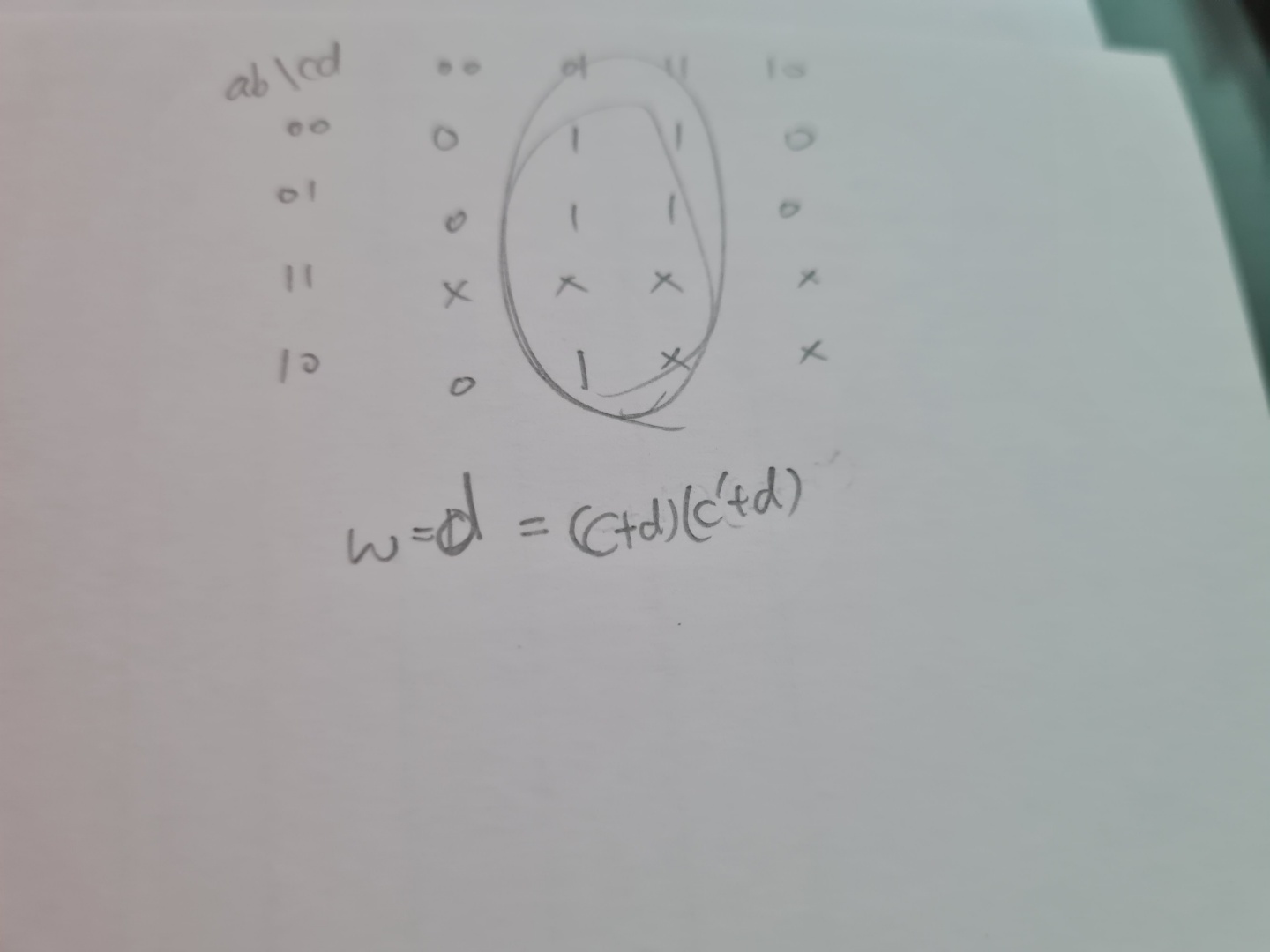
(진리표 작성 및 카로노맵 SOP form, POS form 포함)

8421 형태로 구성된 수를 2421로 구성된 수로 바꾸는 변환 코드를 만들어야 한다. 그래서 입력과 출력 모두 4개로 이루어진다. abcd가 8421의 수이고 xyzw가 2421의 숫자이다. 코드의 각 숫자가 자릿값을 나타낸다. 그래서 2421 코드는 최대 9의 숫자까지만 표현이 가능하다. 카르노 맵을 x,y,z,w에 대해서 하나씩 다 만들어보고 식을 구해주어야 한다.

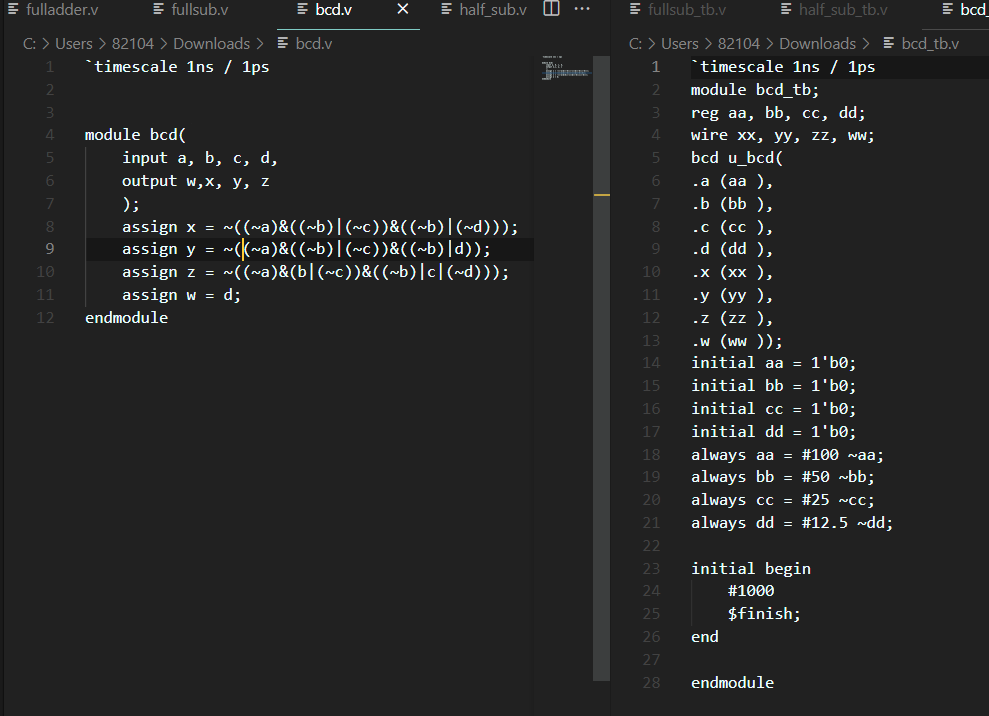


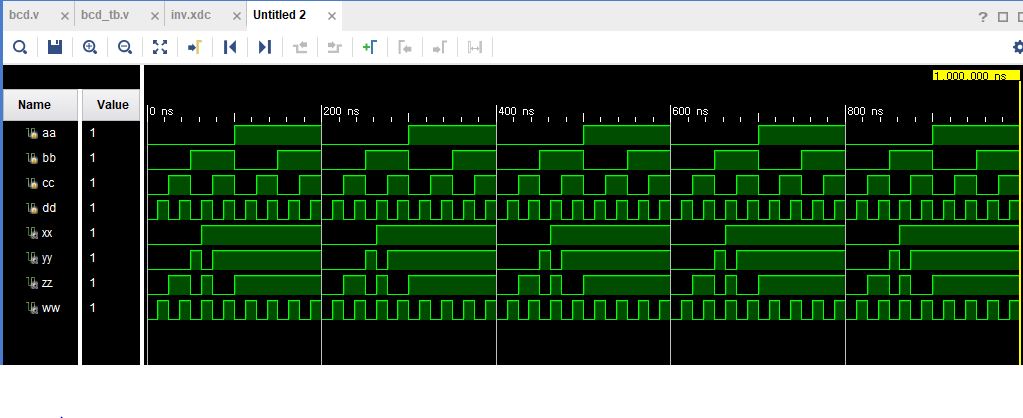






|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8421 (BCD) CODE | | | | | 2421 CODE | | | | |
| a | b | c | d | x | | y | z | w |
| 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | | 1 | 1 | 1 |





5.

처음에는 8421-2421 변환 코드를 카르노맵을 그리지 아니하고 생각해보려고 했는데 규칙을 찾기가 힘들어서 카르노맵을 그려서 해결했다.

6.

-BCD 표기법 : 2진법, 10진법과 같은 디지털 진법체계는 아니고 10진수와 2진수의 변환을 쉽게 하기 위해 생각된 표기 방법이다. 주로 컴퓨터과 사람의 사이에 정보 교환의 징검다리 역할을 한다. 주요 종류로는 가중치 방식 코드, 비 가중치 방식 코드가 있다. 가중치 방식 코드는 8421, 6311 등의 코드가 있고 비교적 단순한 코드 체계이다. 비 가중치 방식 코드는 자리값이 없는 코드라서 3초과 코드, 그레이 코드 등이 여기에 포함된다. 연산의 의미가 없다. 이외에도 EBCDIC 코드가 있는데 extended binary coded decimal interchange code의 줄임말로 BCD 코드를 8비트로 확장시킨 코드이다. 예전 일부 컴퓨터간의 통신용 코드로 이용된 적이 있어서 IBM 360 계열에 해당하는 컴퓨터가 사용했던 적이 있다.