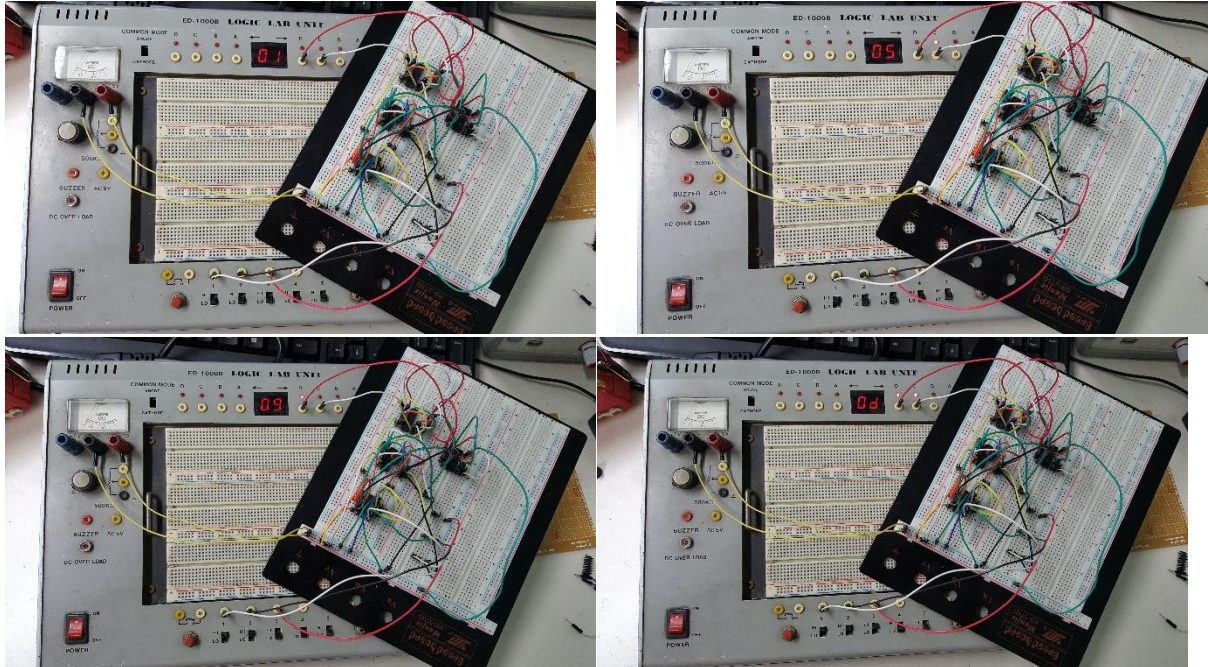


논리설계 2주차 실습 보고서

2017-12751 컴퓨터공학부 이동학



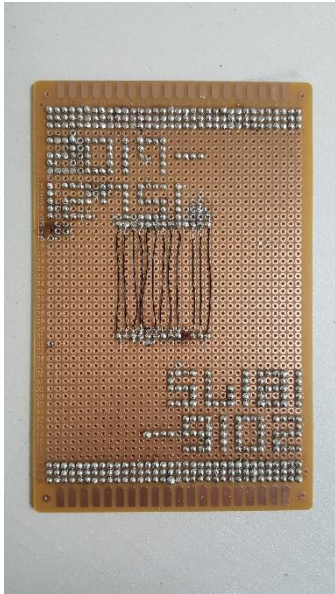
위의 사진 4개는 Full adder를 2-input NAND gates만 이용하여 만든 회로이다. 기존의 Full adder에서 XOR, AND, OR gate를 전부 NAND gate로 바꾸어 구현하였다. 그 결과 16개의 NAND gate가 이용되었지만 나중에 찾아보니 9개의 NAND gate만을 이용하여 만든 최적화된 회로가 있었다. 다음에는 최소한의 gate만을 사용하여 회로를 구현할 수 있도록 노력해야겠다.

Full adder 진리표

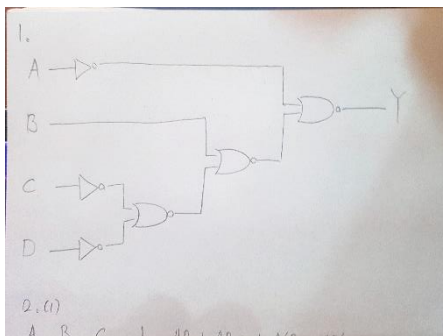
A	B	C _{in}	S	C _{out}
1	1	1	1	1
1	1	0	0	1
1	0	0	1	0
1	0	1	0	1
0	1	0	1	0
0	1	1	0	1
0	0	1	1	0
0	0	0	0	0

공백 1이 0으로
공백 0이 1이 2개 이상
→ C_{out} = 1

Full adder의 진리표이다. 이론상의 입/출력과 실제 입/출력이 일치하는 것을 보아 잘 만든 듯하다.



납땜을 한 결과이다. 처음 해보는 납땜이라 그런지 납이 생각대로 녹고 붙어주지 않아 상당히 고생했다. 또한 전선 피복 벗기는 것이 굉장히 어려웠으며 전선을 판에 납땜 시키는 것도 잘 붙지 않아 어려웠다.



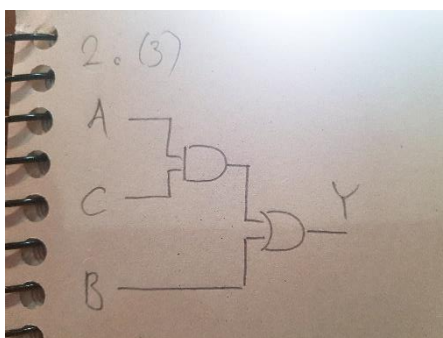
1번 문제의 답이다. AND와 OR을 NOT과 NOR을 이용해서 구현하고 중간에 불필요한 게이트를 삭제했다.

2. (1)

A	B	C	1	$AB + AC + A'B + AB'C$
1	1	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0
0	1	1	1	1
0	1	0	0	1
0	0	1	0	0
0	0	0	0	0

2. (2) $Y = B + AC$

2번에 1번과 2번의 답이다. $(A+A')B = B$ 이고 $(B+B')AC = AC$ 이므로 다음과 같이 바뀐다.



2번에 3번의 답이다. $Y = B + AC$ 이므로 AC를 AND로 연결하고 그 결과를 B와 OR로 연결했다.