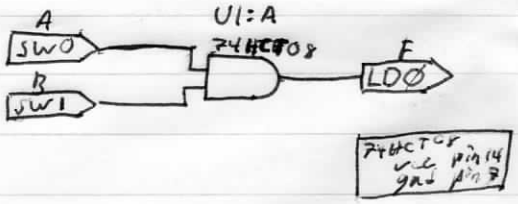
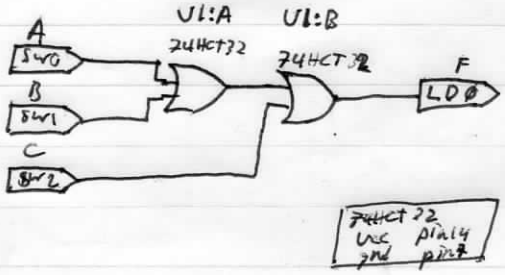
Prac 1 4209435 Justin Mancinelli

1)



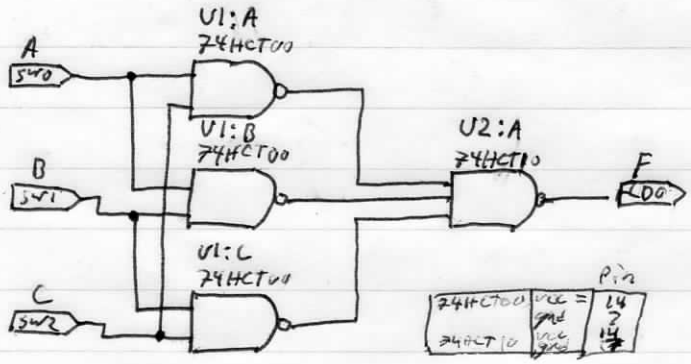
A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

2)



A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

3)



A	B	C	$\overline{AC}$	$\overline{AB}$	$\overline{BC}$	F
0	0	0	1	1	1	0
0	0	1	1	1	1	0
0	1	0	1	1	1	0
0	1	1	1	1	0	1
1	0	0	1	1	1	0
1	0	1	0	1	1	1
1	1	0	1	0	1	1
1	1	1	0	0	0	1

~~Procedure~~

~~1), 2), 3) Familiarized with the equipment.~~

Conclusion (1, 2, 3)

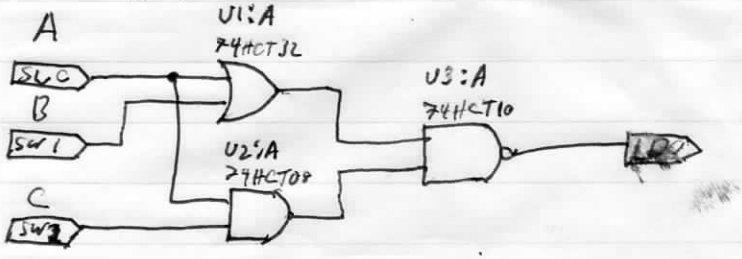
As my first foray into circuit design using bread boards, I have learned that it is important to check and test each step as it is completed. Waiting to debug at the end of design takes longer and is more difficult than debugging at each step.

This lab was quite simple and what I take away from it is a proper understanding of how breadboards, logic workstations, connecting wires and logic gates work together to design computer circuits.

Prac 1

MM 3

Tutor Task



A	B	C	A⊕B	AC	$\overline{(A+B)AC}$
0	0	0	0	0	1
0	0	1	0	0	1
0	1	0	1	0	1
0	1	1	1	0	1
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	1	1	0

74HCT32:	VCC	pin 14
	GND	pin 7
74HCT08:	VCC	pin 14
	GND	pin 7
74HCT10:	VCC	pin 14
	GND	pin 7

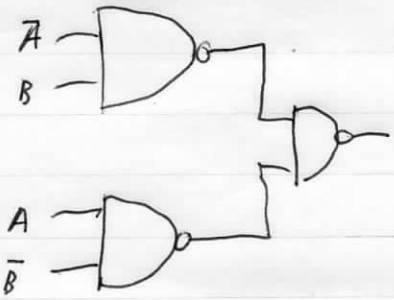
Challenge Task

$A \oplus B = (A \vee B) \wedge \neg(A \wedge B) = \neg(\neg A \wedge \neg B) \wedge \neg(A \wedge B)$

$= \neg\neg(\neg(\neg A \wedge \neg B) \wedge \neg(A \wedge B))$

XOR Implementation with NAND

In Lecture Notes:



NOT: A