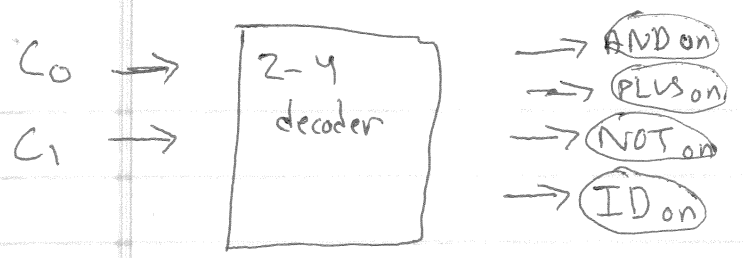
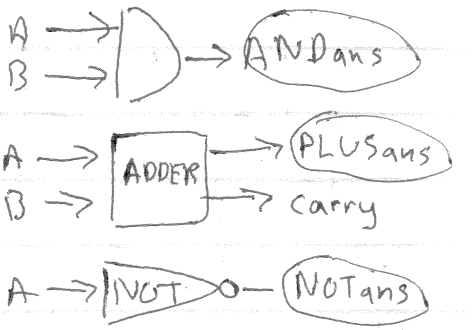
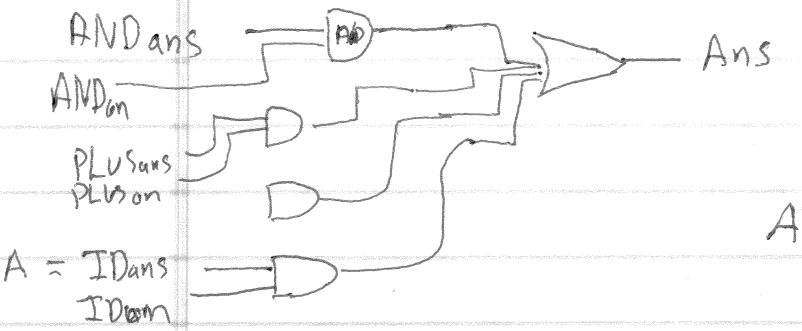
5-2

C = 0, 1, 2, 3
00, 01, 10, 11

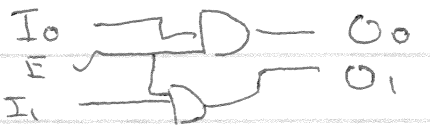
Input = (A), (B), (C)
Output = (Ans), (Test)





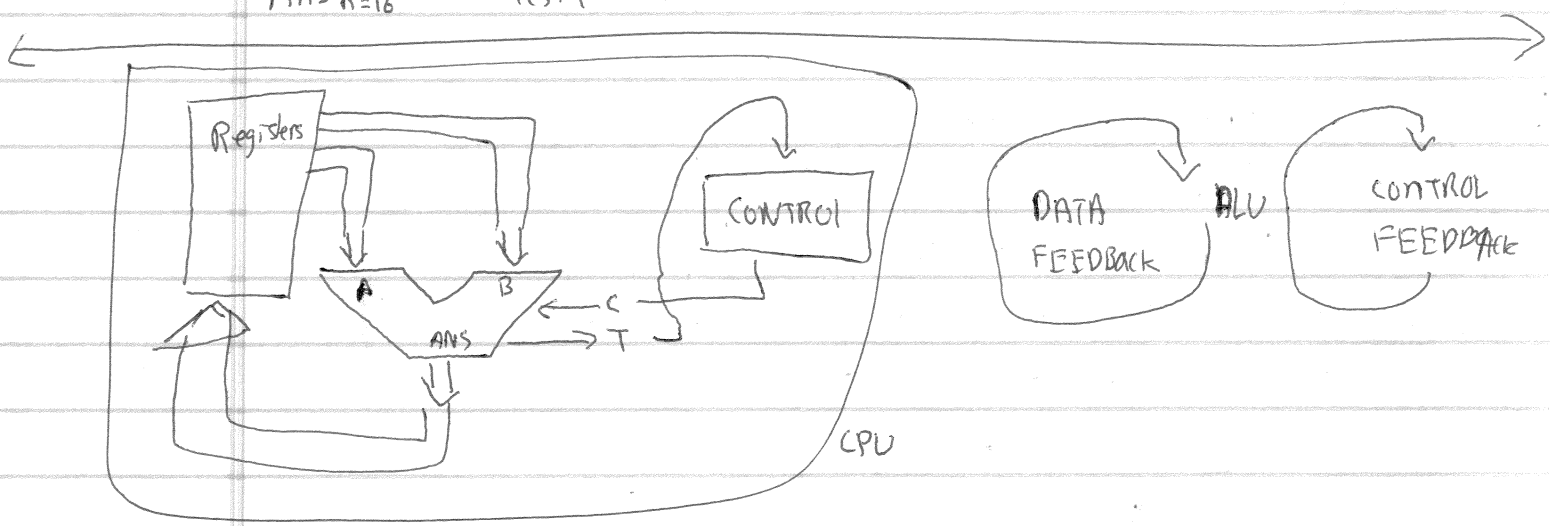


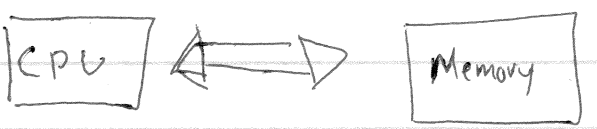
AND' (many wires, one wire) = many



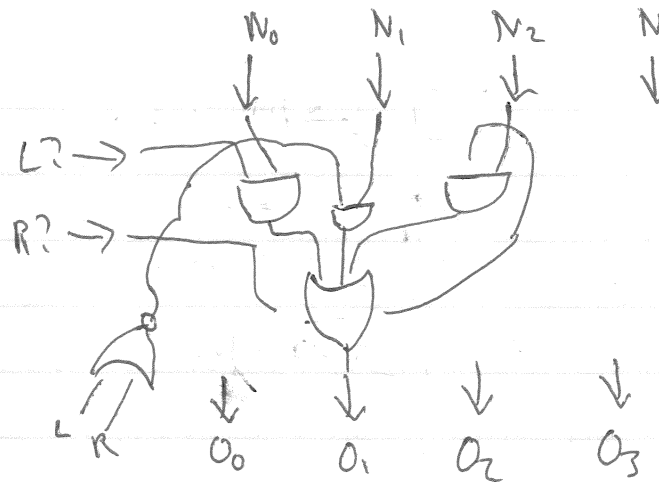
$Test_0 = Z?$

$Ans_{n=16} = Test_1 = N?$





5-3/ Shifter (Left?, Right?, N, O)



NOR

X	Y	Both False
0	0	1
0	1	0
1	0	0
1	1	0

A

< ... , +, LS, +RP ... , PC, PC, +1, ... >

AMUX=A

PC := (PC+1) << 1 (and read)

AMUX=MBR PL := (PC+MBR) << 1

5-1/ Micro-sequence Logic

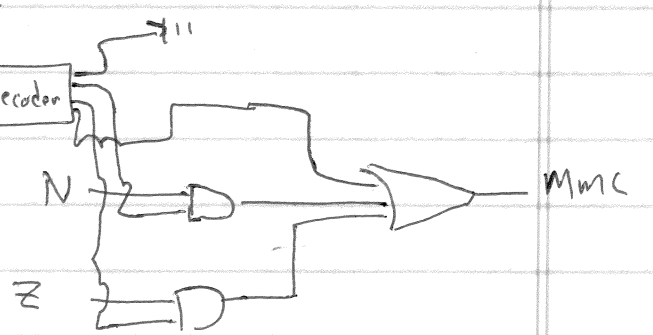
Input: N, Z, COND/2

Output: Mmux-control (0 = +1, 1 = ADDR)

COND = 00 – Never Jump 10 – Jump on Zero

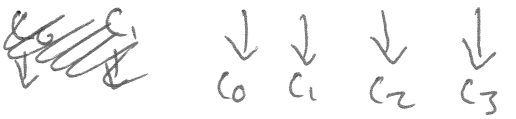
01 – Jump on Neg 11 – Always Jump

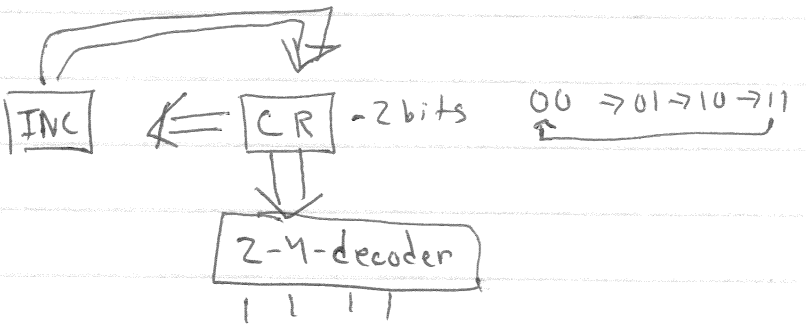
N	Z	COND	Mmux
———		00	0
———		11	1
1	–	01	1
0	–	01	0
–	1	10	1
–	0	10	0



6-2

Clock : N lines (N=4)





Latch (or 1-bit Register)

IN: Data , Enable

OUT: Value

V_{prev}	D	E	V
0	0	0	V_{prev} (0)
1			(1)
V_{prev}	1	0	V_{prev}
–	0	1	0
–	1	1	1