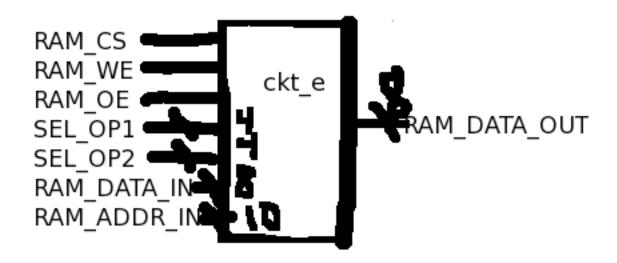
Brian Reber CprE 381 Lab 02

Prelab:

```
entity loader is
    port (big_wank : in std_logic_vector(5 downto 0);
        net_tok : in std_logic;
        tw_cnt : in std_logic;
        clk : in std_logic;
        assert : out std_logic;
        data : out std_logic_vector(9 downto 0);
        addr : out std_logic_vector(7 downto 0);
        spud_out : out std_logic);
end loader;
```



6e)

7) In the entity declaration in part a, there is a missing semicolon at the end of the clk line, and there is also a missing semicolon at the end of the port declaration. In part b, there is a missing closing parenthesis and semicolon for the port declaration.

Ones Compliment	er —	<u> </u>	<u> </u>	<u> </u>	<u> </u>			I	
/ones_complimenter/i	_a 01110111	-		•	10110101		-		
/ones_complimenter/c	_f 10001000	-		•	01001010	-	-		
	0 ns	111111111111111	0 ns	200	ns	300	ns ns	400	ns

Entity:ones_complimenter Architecture:structure Date: Wed Aug 31 02:45:52 PM CDT 2011 Row: 1 Page: 1

т	1 1 1 1 1	
Ones Complimenter Dataflow		
/ones_complimenter_df/i_a 011101111	10110101	
/ones_complimenter_df/o_f 10001000	01001010	
0#\$£0 100 ns	s 200 ns 300 ns 400	ns

/tb_ones_complimenter/s_a	011101110	1110111011	101110111	0111	101101010	111011101	101110111	0111
/tb_ones_complimenter/s_outstructural	100010001	0001000100	010001000	1000	010010101	0001000100	010001000	1000
/tb_ones_complimenter/s_outdataflow	100010001	0001000100	010001000	1000	010010101	0001000100	010001000	1000
					[
0	ns	100	ns	200	ons'	300	ns i	400

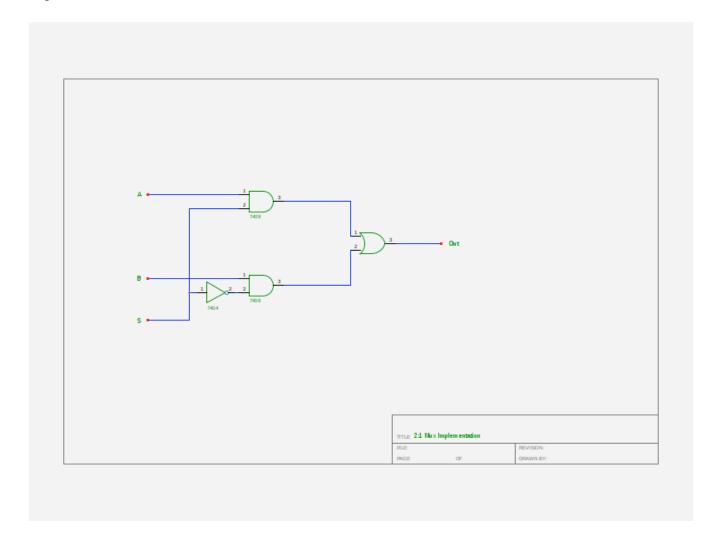
Entity:tb_ones_complimenter Architecture:behavior Date: Mon Sep 05 01:37:08 PM CDT 2011 Row: 1 Page: 1

These are a collection of waveforms for the one's complimenter. From looking at the waveforms, you can tell that it is behaving as expected (it is properly inverting bits).

Two-Input Mux

S	A	В	Out
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Expression: $S * A + \sim S * B$



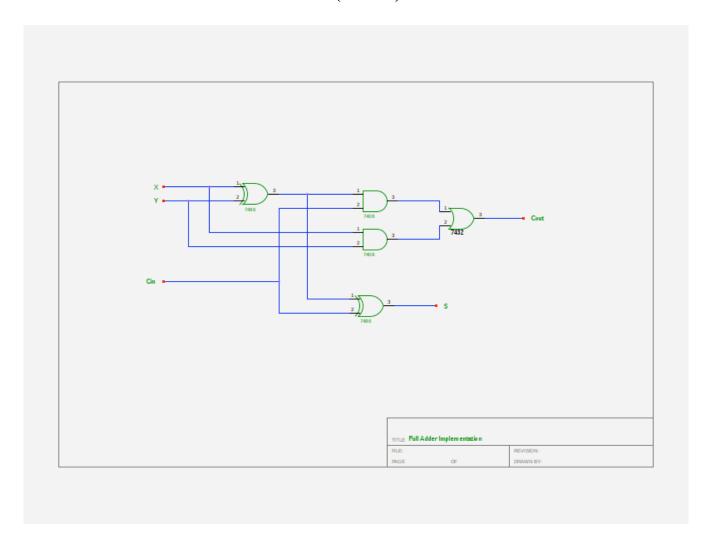
/tb_two_to_one_mux/s_a	011101	1101110	111011	101110	110111				-				l
/tb_two_to_one_mux/s_b	100010	001000	000100	0100010	0001000				101010	101010	010		ı
/tb_two_to_one_mux/s_outstructura	011101	1101110	1111		100010	0010001	000		101010	101010	010		l
/tb_two_to_one_mux/s_outdataflow	011101	1101110	1111		100010	010001	000		101010	101010	010		l
/tb_two_to_one_mux/s_s									-		•		ı
													ı
													ı
													l
													ı
													ı
													ı
													ı
													ı
													ı
													l
													l
													l
													l
													ı
													ı
	HIIIIIIII Ins	<u> </u>		<u> </u>			[l				[]	Lii

Entity:tb_two_to_one_mux_Architecture:behavior_Date: Mon Sep 05 03:51:31 PM CDT 2011 Row: 1 Page: 1

Full Adder

X	Y	Cin	Cout	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

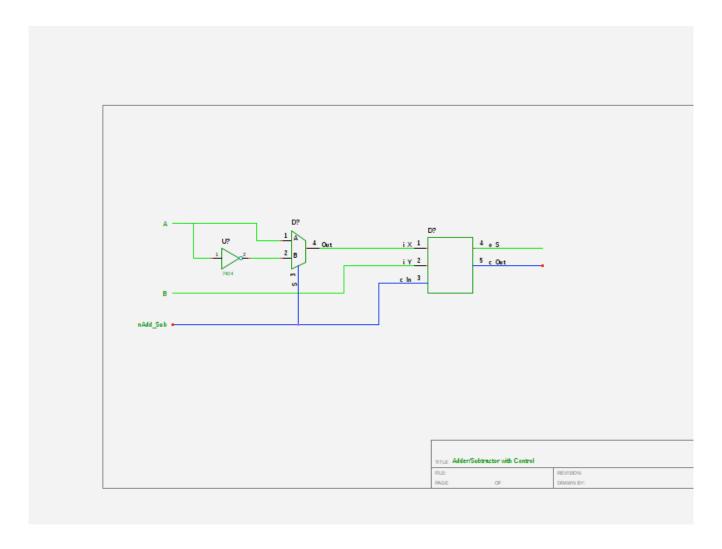
Expression: $S = \sim X * \sim Y * C + \sim X * Y * \sim C + X * \sim Y * \sim C + X * Y * C = X \text{ xor } Y \text{ xor } C$ Cout = Y * C + X * C + X * Y = (Y xor X) * C + X * Y



/tb_full_adder/s_x	01110	11101	11011	10111	1110	11011	1		11111	11111	11111	11111	11111	11111	1		
 /tb_full_adder/s_y		1	L	1 .	01110		ł	1	1		00000			!	00000	D	
/tb_full_adder/s_outstructural				1 .	11101	11011	10111	p	1		00000		00000	00000	00000	D	
/tb_full_adder/s_outdataflow					11101	11011	10111	D	00000	00000	00000	00000	00000	000000	0		
/tb_full_adder/s_cin																	
/tb_full_adder/s_coutstructural																	
			, ,						l				l	l <u>.</u> .	!		
0	ns I			200	ns '			40	ons '			600	ns '			, 800	n

Entity:tb_full_adder Architecture:behavior Date: Mon Sep 05 05:25:51 PM CDT 2011 Row: 1 Page: 1

Adder/Subtracter with Control



This design uses an N-bit Adder, an N-bit 2:1 Mux and an N-bit inverter. If $nAdd_Sub = 0$, the mux will provide A to the adder, making it perform A + B. If $nAdd_Sub = 1$, the mux will give the adder the inverse of A (by using the N-bit inverter). It will also pass along 1 to the adder as the carry in bit, which causes it to perform the equivalent of A - B.

/tb_add_sub/s_x	0111011	1011101	1101110	1110111	0111				1111111	1111111	11	
/tb_add_sub/s_y	0000000	0000000	00		0111011	1011101	11		0000000	0	0000000	0
/tb_add_sub/s_outstructural	0111011	1	1000100	0	1110111	0	0000000	0	0000000	0	0000000	0
/tb_add_sub/s_nadd_sub												
/tb_add_sub/s_cout												
0	ns	200	,,,,,,,,,,) ns	400		600	,,,,,,,,,,) ns	800) ns	100	,,,,,,,,,, 0 ns	1200

Entity:tb_add_sub_Architecture:behavior_Date: Mon Sep 05 06:11:09 PM CDT 2011 Row: 1 Page: 1

In this waveform, I have tested multiple values for the adder/subtracter unit. I tried to make test cases which would test different aspects of addition and subtraction including adding a positive and a negative number, subtracting a value from itself, adding two positive values, adding to make it overflow. See the tb_add_sub.vhd file for all of the values used, and their expected results.