Exam 1 EECE 338 Spring 2010

Name:

Question	Possible	Score
1	10	8
2	15	6
3	15	13
4	15	8
5	15	12
6	15	6
7	15	14
Total	100	67

General information that may be useful sometime during test:

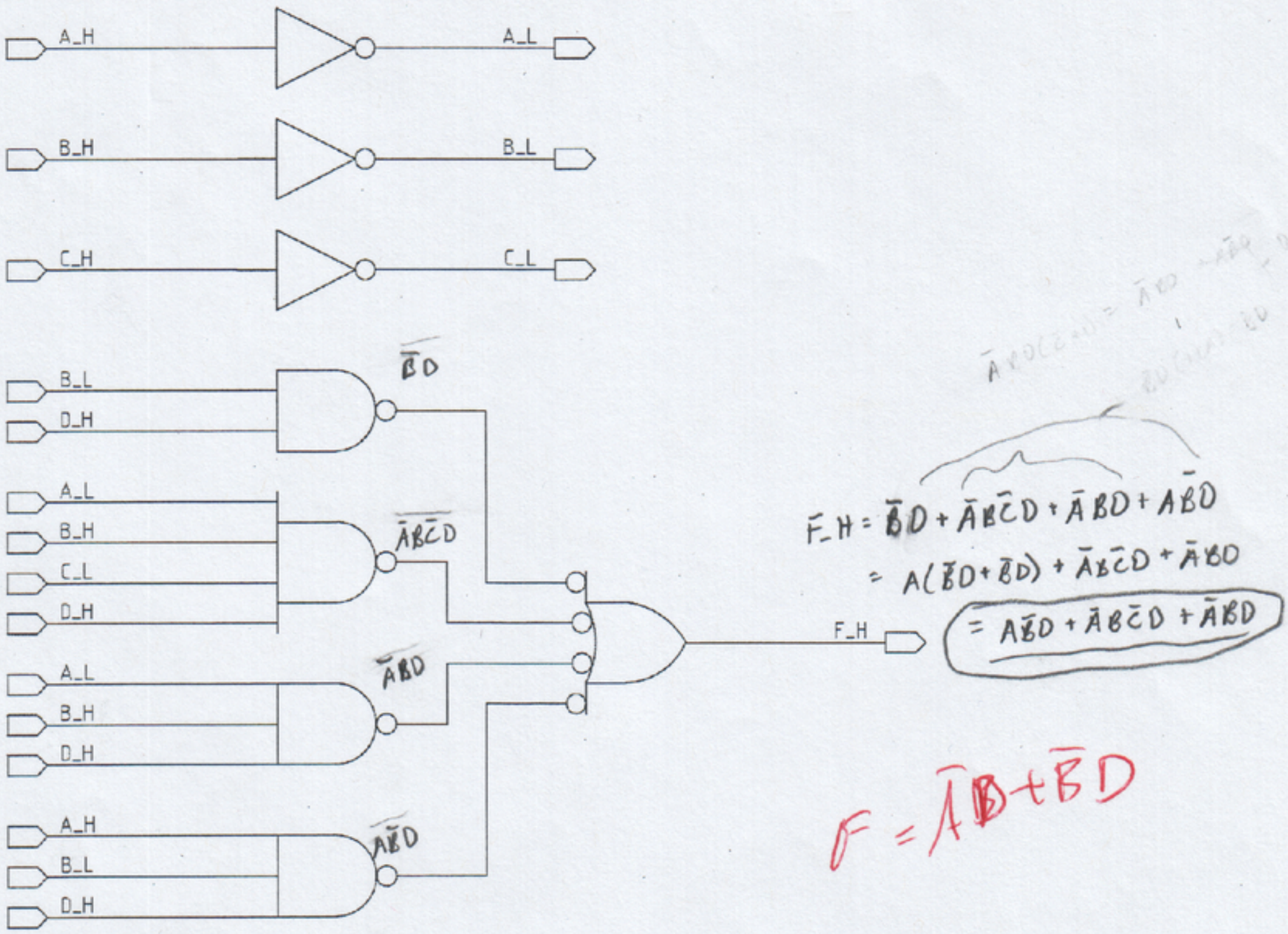
Table of Powers of Two

N	2^N		N	2^N		N	2^N		N	2^N
0	1		8	256		16	65,536		24	16,777,216
1	2		9	512		17	131,072		25	33,554,432
2	4		10	1,024		18	262,144		26	67,108,864
3	8		11	2,048		19	524,288		27	134,217,728
4	16		12	4,096		20	1,048,576		28	268,435,456
5	32		13	8,192		21	2,097,152		29	536,870,912
6	64		14	16,384		22	4,194,304		30	1,073,741,824
7	128		15	32,768		23	8,388,608		31	2,147,483,648

1. The general representation question: Give the correct pattern (in binary, hex, etc.) for the following numbers:

Coding method	Base 10 value	Answer
Unsigned Binary (10 bits)	199	0011 0 0 0 1 1 1 ✓
Two's Complement (8 bits)	-95	0 1 0 1 1 1 1 1 1 0 1 0 0 0 0 1
Two's Complement (10 bits, 5 fractional bits)	-8.375	0 1 0 0 0 . 0 1 1 0 0 0 + 1 0 1 1 1 . 1 0 0 1 1 1 1 0 1 1 1 . 1 0 1 0 0 0
Unsigned Binary (8 bits)	215	~~1 1 1~~ 0 1 ~~4~~ 1

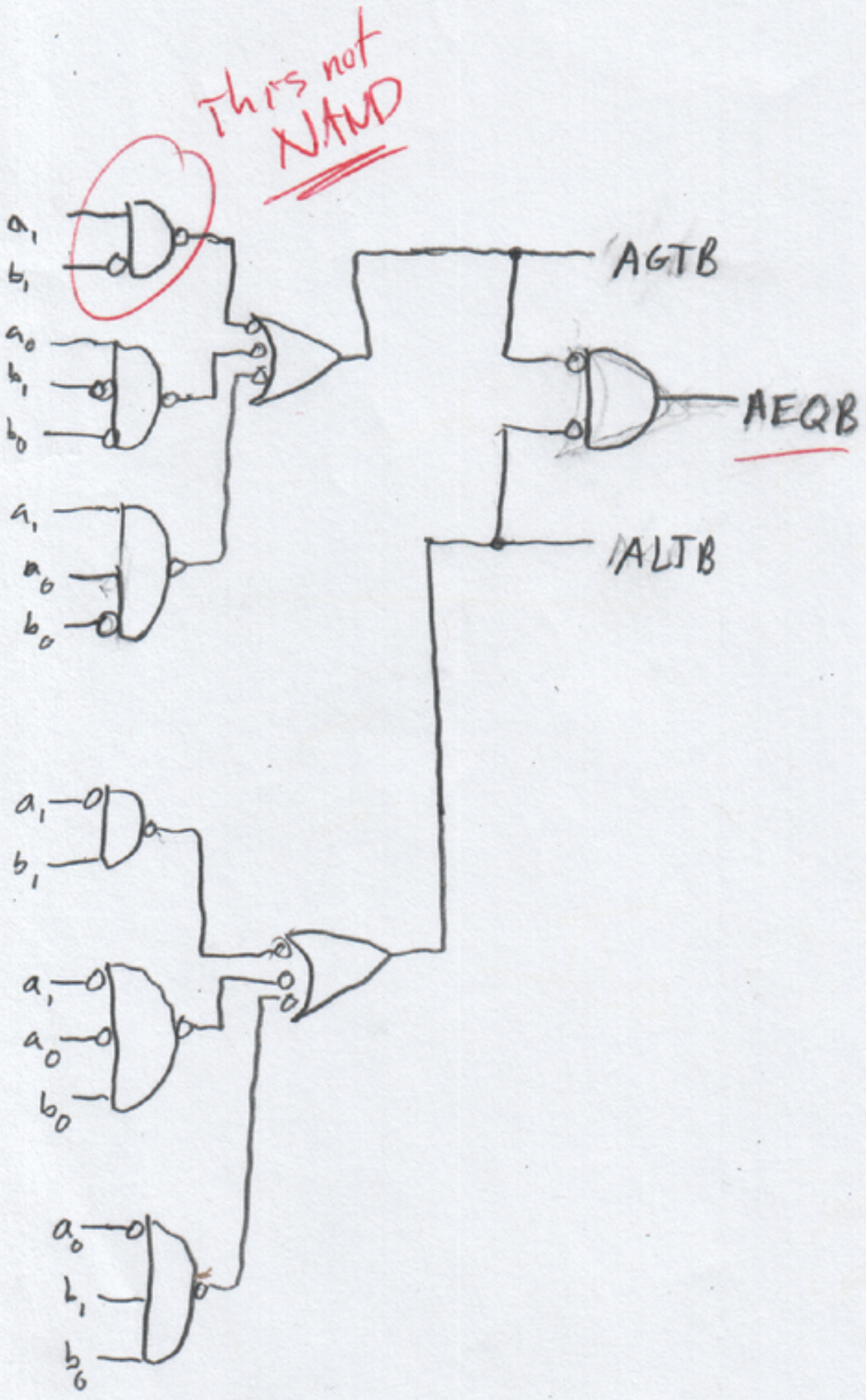
2. Give the logic equation for the following gate system. The system is obviously – given the history of the sadistic instructor – much simpler than the gates shown. There is really only one place where tricky-ness enters the picture: including terms that aren't needed 'cuz the work is done by other gates….

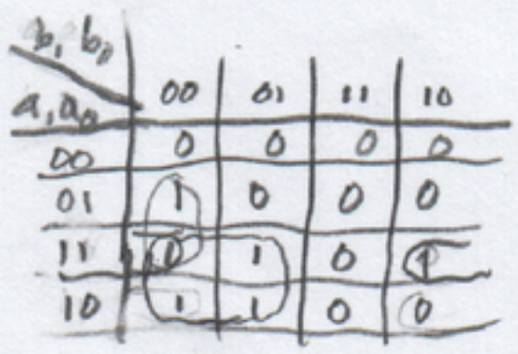


3. The Combinational Design Question: In the space provided below, design a two-bit comparator. That is, consider a system that has four bits of input: two bits for unsigned binary A value (A1,A0) and two bits for unsigned binary B value (B1,B0). The system has three outputs: AGTB_H (A greater than B, asserted high), ALTB_H (A less than B, asserted high), and AEQB_H (A equal to B, asserted high) Complete the table and carry out the design steps to create a gate level representation of the system. Use only NAND gates and inverters. Oh, you can use a single 2-input NOR gate if you would like. Remember to show the correct gate shapes for the functions performed.

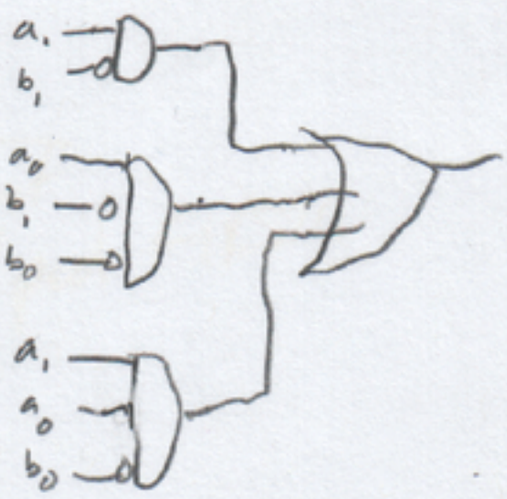
ALTB AEQB

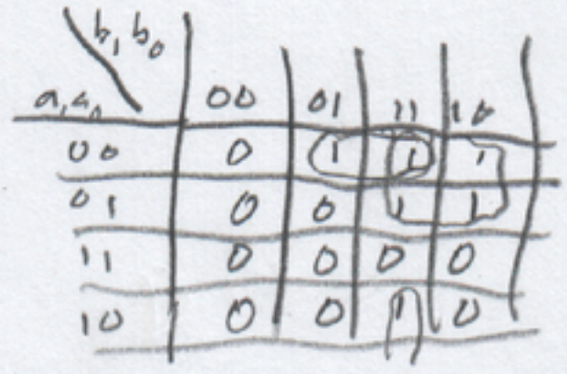
A1	A0	B1	B0	AGTB	~~AEQB~~	~~ALTB~~
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	1	0	0
0	1	0	1	0	0	1
0	1	1	0	0	1	0
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	0	1	0
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	0	1





$AGTB = a_1\bar{b}_1 + a_0\bar{b}_1\bar{b}_0 + a_1a_0\bar{b}_0$





$ALTB = \bar{a}_1b_1 + \bar{a}_1\bar{a}_0b_0 + \bar{a}_0b_1b_0$

4. The VHDL Syntax Question: Identify and correct the errors in the VHDL code:

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;

entity EXAM_QUES4 is
  port (
    X1 : in  STD_LOGIC;
    X2 : in  STD_LOGIC;
    C  : in  STD_LOGIC;
    Z2 : out STD_LOGIC;
    Z3 : out STD_LOGIC;
  );
end entity EXAM_QUES4;

architecture FIRST1 of EXAM_QUES_4 is

signal SAL-D : STD_LOGIC;
begin

  X2 <= SAL-D;
  process ( X1, X2, Z3 ) is
  begin
    Z3 <= Z2 and X1 or X1 and C;
    if C = '1' and X1 = '1' then
      Z2 <= X1;
    else if X2 = '0' then
      Z2 <= not X1;
    end if;
    end;
  SAL_D <= '1' if X2 = X1 else 'Z';
end architecture EXAM_QUES_4;
```

improper name need underscore => SAL_D

SAL_D;

Z3

Need brackets

elsif

end if;

end process;

'1' (single quotation marks)

SAL_D

<=

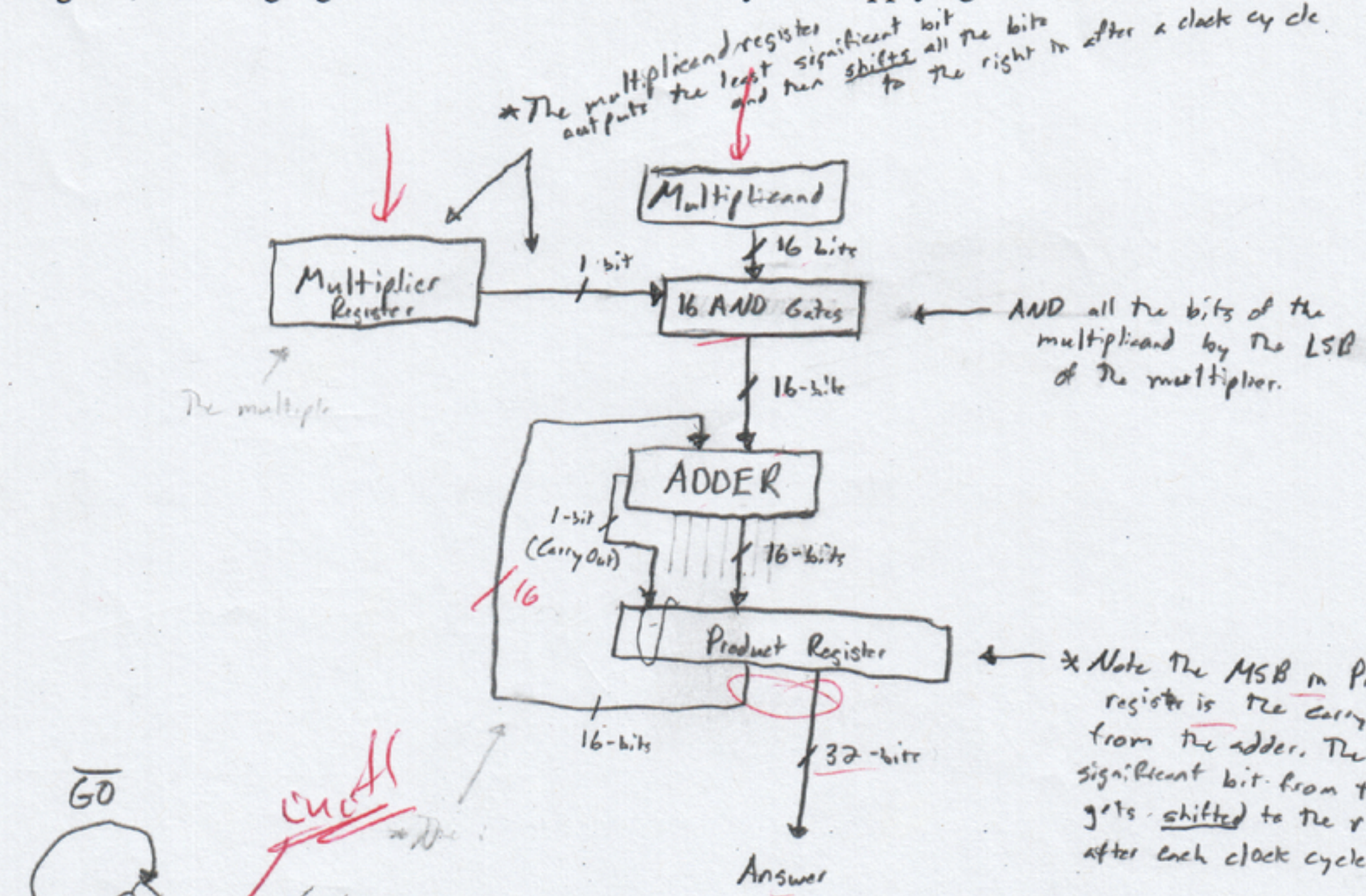
=

but, it won't work here

don't need this

yes, but could put FIRST1

5. Multiply question: Show a detailed block diagram of a multiplier that works using the simplest grade-school algorithm. Assume a 16 by 16 multiply. Show where the shift(s) are to be performed. Give a state diagram that identifies the work to be done by your block diagram, assuming a go/done interaction with the system supplying the data.



6. Floating Point question: Consider a floating point system that is a small version of the IEEE floating point system. That is, it has all the fields, and the same characteristics, but it is smaller. This system has a sign bit, a four bit exponent field, exponent is stored in excess-7 format, and a seven bit mantissa field, mantissa is a number between 1 and almost 2, and mantissa utilizes hidden-bit concept. Also, the all 1's pattern for the exponent is saved for special stuff, and the all 0's pattern is for un-normalized numbers. Ignore un-normalized values and special stuff for this question. For this system:

a) what is the smallest representable non-zero, positive value?

Bit Pattern	Value
0 0111 0000000	$(-1)^0 * 2^0 * 1.0 = 1$

b) what is the largest representable value?

Bit Pattern	Value
0 1110 1111111	$2^8 * \frac{255}{128} = (256 \times 2) = 512$

c) how many values can be represented in all?

Number of Values
$2^{12-1} = 2^{11} = 2048$

d) what is the value of 111000110000?

Value
$-2^5 * (-1.375) = -44$

e) represent $3\ ^{9}/_{16}$ in this system

Bit Pattern
0 1000 1100100

7. Math Question 1 (addition): In the space below, develop the equations for the generate and propagate functions for a two-bit adder. That is, assume that a two-bit adder has inputs A (represented as A_1,A_0) and B (represented as B_1,B_0) and CIN. You are to provide the equations for generate and propagate for this two-bit system. Remember that for a full adder, COUT = A•B + A•CIN + B•CIN.

$= AB + CIN(A+B)$

Generate Propagate

$C_0 = G_0 + C_i P_0$

$C_1 = G_1 + C_0 P_1 = G_1 + P_1 G_0 + C_i P_1 P_0$

Generate Propagate

Propagate $= P_1 P_0 = (A_1+B_1)(A_0+B_0) =$

Generate $= G_1 + P_1 G_0 = A_1 B_1 + (A_1+B_1)(A_0 B_0) = A_1 B_1 + A_1 A_0 B_0 + A_0 B_1 B_0 =$