

Name:

Problem	Possible	Score
1	15	15
2	25	23
3	25	16
4	15	8
5	20	20
Total	100	82

Please write legibly!

1. General information question:

- a) Consider a fixed point two's complement system that uses 9 bits for representation. What is the value represented by 011100.101? What is the value represented by 100111.010? Note the radix point located three bit positions from the right edge.

$$\begin{array}{r} 011100.101 = 28.625 \\ 100111.010 = -24.75 \end{array}$$

$$\begin{array}{r} 000011.010 \\ 100111.010 \\ +111000.101 \\ \hline 011100.110 \end{array}$$

↓
6 8 4 2 1

- b) What are the three control lines (note: not address or data) involved in the standard 4 event protocol?

Request, acknowledge, read/write

- c) In the asynchronous sequential design method we have examined, changes in inputs move the state of the system in which manner in the final state table/K-map? And changes in the feedback variables influence the state of the system how?

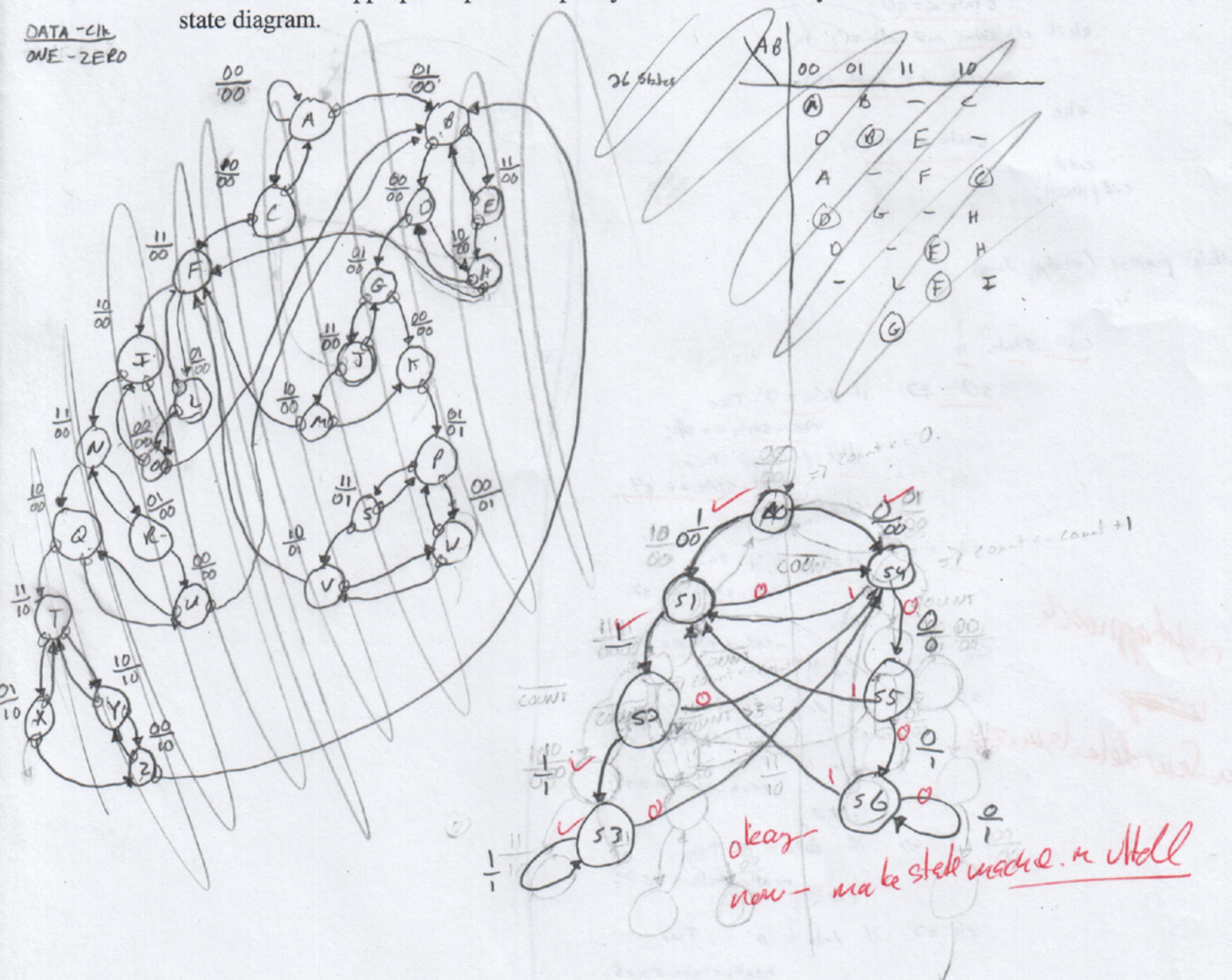
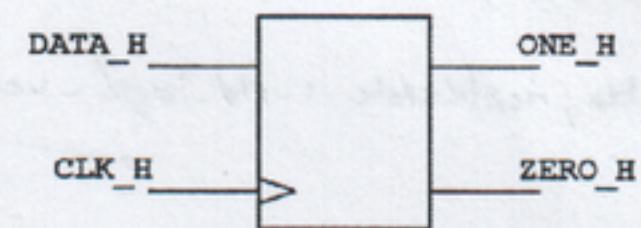
Inputs move across columns/horizontal adjacent state movements
 while feedback variable influence row/vertical adjacent state transitions.

- d) In the small space below, provide VHDL code for a process to implement a D-type flop-flop with inputs D (data), CLK_H (clock), and RST_H (clear), and output Q (data out). The clear is asynchronous. There is no 'set' function. Active edge of clock is low-to-high transition.

```
process (D, clk-h, rst-h) is
begin
  if rst-h = '1' then
    D <= '0';
  elsif clk-h = '1' and clk'event then
    Q <= D;
  end if;
end process;
```

Review Edge

2. Sequential system design #1: Consider the system whose block diagram is shown at the right. Shown are clock and data inputs; assume that there is also an input named `RST_H` (for the reset). Also shown are two outputs: `ONE_H` and `ZERO_H`. The function of the system is as follows: whenever the input data string has three or more '1' values in a row, then the output `ONE_H` will be asserted. Whenever the input data string has three or more '0' values in a row, then the output `ZERO_H` will be asserted. Otherwise, neither output is asserted. Remember that the state diagram gives a graphical representation of the desired behavior of the system... So, for this system, provide a) a state diagram and b) a VHDL entity-architecture pair that implements the system. Use a case construct in the appropriate place to specify the work. Hint: my solution has seven states in state diagram.



```
signal s0,s1,s2,s3,s4,s5 : std-logic;
```

```
signal state,next-state : std-logic-vector(2 downto 0);
```

```
next-state: process (clk,rst-h,state)
```

```
begin
    if rst-h = '1' then
        state <= s0;
    elsif clk'event and clk = '1' then
        state <= next-state;
    else
        state <= state;
    endif;
end process;
```

```
next-state: process (state,data)
```

```
begin
```

case state is

s0 => if data = '1' then
 next-state <= s0;
else data = '0' then
 next-state <= s4;
endif;

s1 => if data = '1' then
 next-state <= s2;
else
 next-state <= s4;
endif;

s2 => if data = '1' then
 next-state <= s3;
else
 next-state <= s4;
endif;

s3 => if data = '0' then
 next-state <= s2;
endif;

s4 => if data = '0' then
 next-state <= s5;
else
 next-state <= s2;
endif;

s5 => if data = '0' then
 next-state <= s6;
else

s6 => if data = '1' then
 next-state <= s1;
endif;

end process;

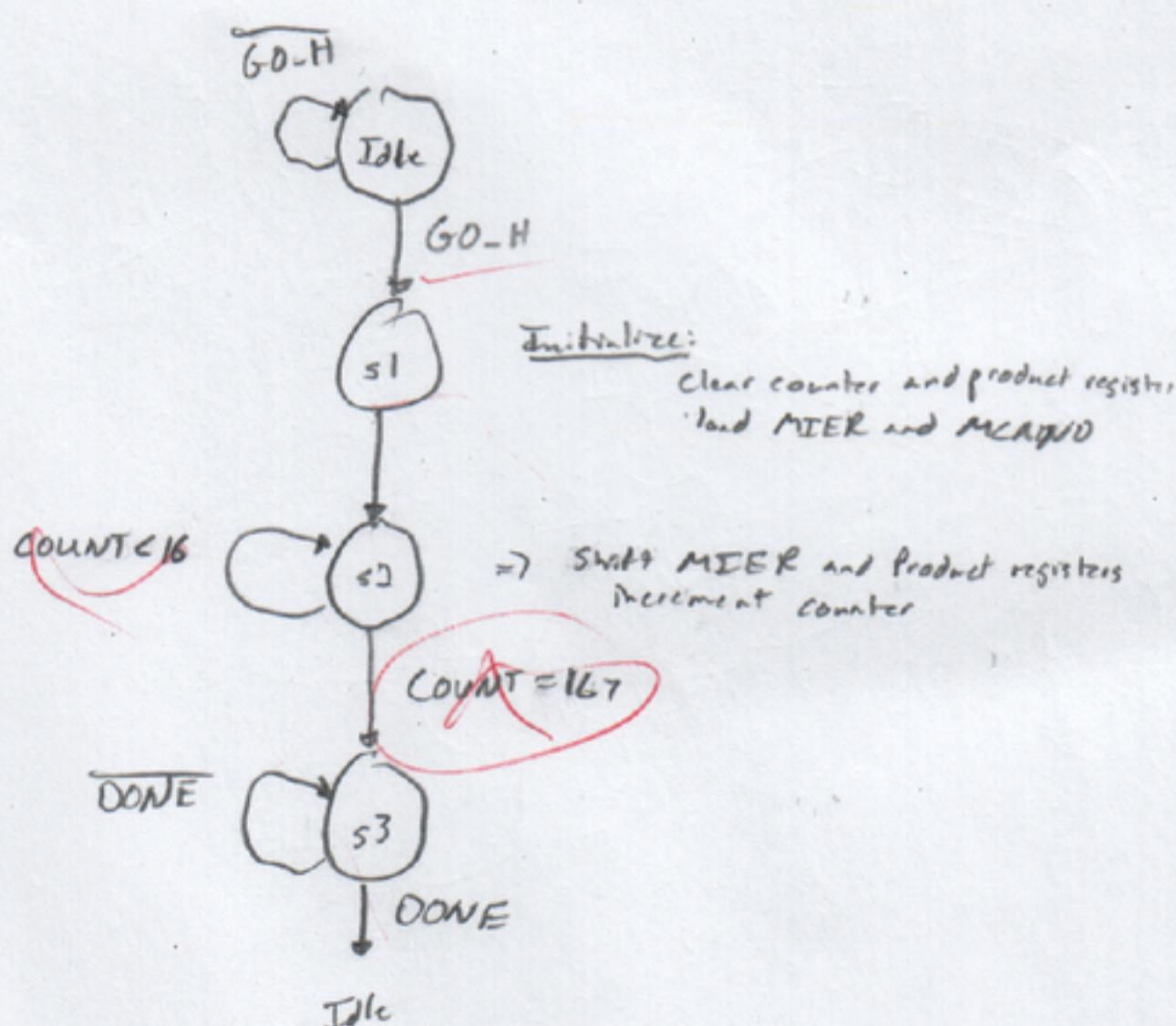
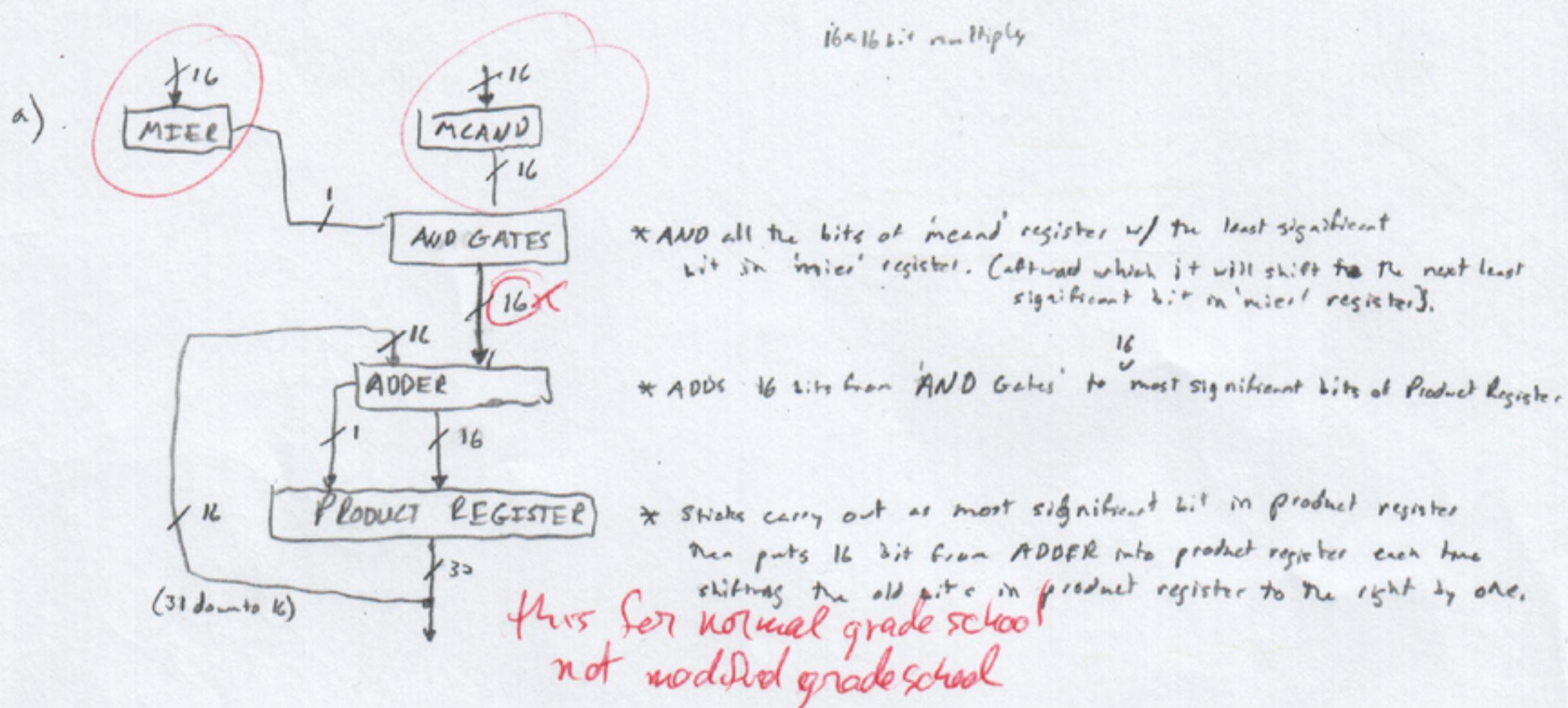
ONE-H <= '1' when state = s3 else '0';

ZERO-H <= '1' when state = s6 else '0';

met approach

~~a few details missing~~

3. Sequential system design & math stuff: In the space provided below, present two things. Part (a) is a block diagram for a multiplier that uses the modified-grade-school algorithm. This multiplier should work the partial product array most-significant to least-significant. The data path needs to include whatever logic is needed to stop when no more values are available for adding to the product. Indicate on the diagram the various registers and data paths, as well as the width of those registers and data paths. For those registers that must shift, identify the shift direction. For Part (b), give a state diagram for correct operation of the multiplier, using the GO-DONE handshake method from user to multiplier.



4. Now for the math question itself. We spent some time working on the Booth Algorithm method of doing work for multiplying two signed numbers. Using this algorithm, show how to multiply two 5-bit patterns together to get the correct result. The patterns are A = 10111 (which is -9) and B = 11001 (which is -7).

$$\begin{array}{r}
 & 10111 \\
 & \underline{11001} \\
 \hline
 S_0 = D - B_0 & 2 + 7 = 9 = 10111 \\
 & , 0000000 \\
 & , 00000000 \\
 & , 000000000 \\
 A'_0 = A * S_0 & -1(9) = \underline{\underline{10111000}} \\
 & , 01110000 \\
 \hline
 B'_0 = C * S_0 & -7(9) = \underline{\underline{1000001111}} \\
 & , 328 \in 4 \Rightarrow 1
 \end{array}$$

$$R_2' = R_2 \times \bar{g}_1 = -q(65) =$$

If it were a
zero I believe
you would take
the two's conflict

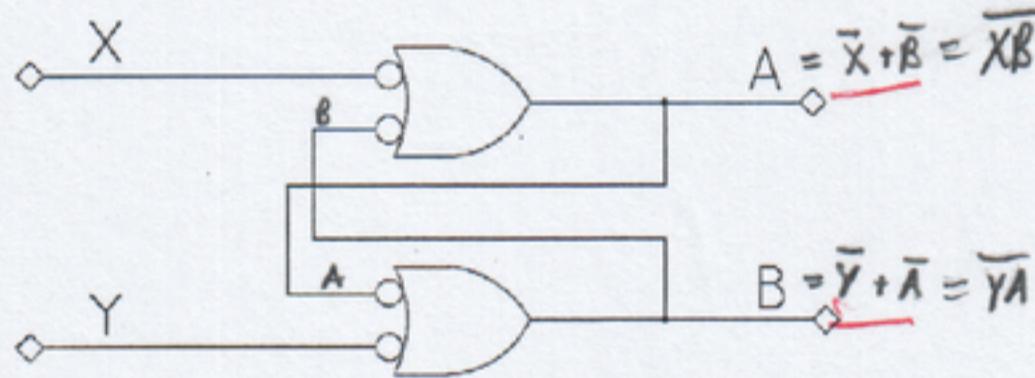
$$\begin{array}{r} -9 \\ \times -7 \\ \hline 63 \end{array}$$

Running sum

	0	0	0	0	0	
	1	0	1	1	0	
	0	0	0	0	0	
	0	1	0	1	1	0
	0	0	0	0	0	0
	0	0	1	0	1	1
	0	0	0	0	0	0
	0	0	0	1	0	1
	1	0	1	1	1	0
	0	1	1	0	0	1
	1	0	1	1	1	0
①	0	0	0	1	1	1

63

5. Asynchronous logic question. This question is intended to check your understanding of analysis of asynchronous systems. Remember that analysis is the reverse order of the steps of synthesis. Below are located a logic diagram and a K-map/final table. In the synthesis of circuits, the gates are the final step; here they are the first step. This particular system is simply two cross-coupled NAND gates, with A the name of one feedback variable and B the name of the other. So, for the question. In the K-map/final table provided, label the groups appropriately (which rows/cols are for which variables?) and then show the excitation table given by the gates, and identify stable states. Finally, show a state diagram for the system.



		X	Y			
		00	01	11	10	
AB		00	A	D	F	C
A	00	11	11	11	11	
	01	11	11	01	01	
B	11	11	01	00	01	
	10	11	10	10	11	

