Please enter your name, Athena login name, and recitation section above. Enter your answers in the spaces provided below. Show your work for potential partial credit. You can use the extra white space and the backs of the pages for scratch work.
Problem 1. Static Discipline (14 points)

Ben Bitdiddle has been given a hypothetical Device F and told to come up with signaling specifications for it. Consider the voltage transfer characteristic (VTC) of Device F shown below (not to scale). Ben is considering two different choices for each of $V_{OH}$ and $V_{OL}$, as indicated by the dashed lines on the graph below. The precise values of these thresholds are listed in the table. $V_{IH}$ is known to be 4.6V.

![Graph showing voltage transfer characteristic with dashed lines for $V_{OH}$ and $V_{OL}$ choices.]

(A) (4 points) For each choice of $V_{OH}$, circle YES if it both follows the static discipline and allows for a positive **high noise margin**, or circle NO if it does not satisfy both of these conditions. If the answer depends on knowing values that are not provided, then circle DON’T KNOW. Explain each of your answers.

Explaination:

$V_{OH,A}$ (circle one): YES NO DON’T KNOW

Explaination:

$V_{OH,B}$ (circle one): YES NO DON’T KNOW
(B) (4 points) $V_{IL}$ is not known right now but is guaranteed to be between 1.5 V and 2.5 V. In addition, all valid low inputs are guaranteed to produce valid high outputs. For each choice of $V_{OL}$, circle **YES** if it both follows the static discipline and has a positive low noise margin, or circle **NO** if it does not satisfy both of these conditions. If the answer depends on knowing values that are not provided, then circle **DON’T KNOW**. Explain each of your answers.

<table>
<thead>
<tr>
<th>$V_{OL}$, C (circle one):</th>
<th>YES</th>
<th>NO</th>
<th>DON’T KNOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$V_{OL}$, D (circle one):</th>
<th>YES</th>
<th>NO</th>
<th>DON’T KNOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(C) (3 points) Suppose Ben settles on the following signaling specification which follows the static discipline:

- $V_{IL} = 1.5$ V
- $V_{IH} = 4.7$ V
- $V_{OL} = 1.1$ V
- $V_{OH} = 4.9$ V

Calculate the high and low noise margins as well as the noise margins of the device as a whole.

- **High Noise Margin:** ________________ V
- **Low Noise Margin:** ________________ V
- **Overall Noise Immunity:** ________________ V
(D) (3 points) Alyssa P. Hacker has another Device G with the following VTC that she would like to use with Device F. Can she use Device G with the signaling specifications described in part (C)? If she can, circle NO CHANGES and explain why no changes are needed. Otherwise, circle CHANGES NEEDED and change exactly one of the thresholds of the signaling specification to a new value such that Device G can be used while obeying the static discipline and maximizing noise margins. Keep in mind that VTCs may touch the edge of, but not enter the forbidden region.

\[ V_{\text{out}} \]

\[ (1.5, 5.0) \]

\[ (4.6, 1.1) \]

\[ V_{\text{in}} \]

Are changes needed? (circle one): NO CHANGES  

CHANGES NEEDED

If No Changes:

Provide explanation: ________________________________

If Changes Needed:

Threshold to Change (circle one): V_{IL}  V_{IH}  V_{OL}  V_{OH}

Value to change to: ______________________ V
Problem 2. Boolean Algebra and Combinational Logic (11 points)

(A) (2 points) Consider the logic diagram below, which takes 4 inputs \{a,b,c,d\} and computes two outputs \{x,y\}. Using the $t_{PD}$ information for the gate components shown in the table below, compute the $t_{PD}$ for the circuit.

\[
\begin{array}{|c|c|}
\hline
\text{Gate} & \text{$t_{PD}$} \\
\hline
\text{XNOR2} & 5.5\text{ns} \\
\text{AND2} & 3.5\text{ns} \\
\text{OR2} & 2.5\text{ns} \\
\text{INV} & 1.0\text{ns} \\
\hline
\end{array}
\]

\[t_{PD} (\text{ns}) = \]
(B) (3 points) Find the normal form expression and a minimal sum-of-products expression for output \( F \) of the circuit described by the truth table shown below.

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Normal form for \( F \) = 

Minimal sum of products for \( F \) = 
(C) (6 points) Simplify the following Boolean expressions by finding a minimal sum-of-products expression for each one. (Note: These expressions can be reduced into a minimal SOP by repeatedly applying the Boolean algebra properties we saw in lecture.) Make sure that your final answer is in **minimal sum-of-products** form.

1. \( \bar{x} + \bar{x}z + \bar{y}z \)

2. \( xy\bar{z} + x \)

3. \( x\bar{y} + \bar{y}z(x + \bar{z}) \)
Problem 3: CMOS Logic (16 points)

For each of the circuits shown below, please fill out the truth table that the circuit implements. Then provide a logical function for Out in terms of its inputs (a, b, c) or (a, b). Assume that you are given all inputs $a, \overline{a}, b, \overline{b}, c$, and $\overline{c}$.

(A) (4 points)

\begin{align*}
\text{Out}(a, b, c) &= \text{___________________________} \\
\begin{array}{ccc|c}
 a & b & c & \text{Out} \\
 0 & 0 & 0 & \\
 0 & 0 & 1 & \\
 0 & 1 & 0 & \\
 0 & 1 & 1 & \\
 1 & 0 & 0 & \\
 1 & 0 & 1 & \\
 1 & 1 & 0 & \\
 1 & 1 & 1 & \\
\end{array}
\end{align*}

(B) (4 points)

\begin{align*}
\text{Out}(a, b) &= \text{___________________________} \\
\begin{array}{cc|c}
 a & b & \text{Out} \\
 0 & 0 & \\
 0 & 1 & \\
 1 & 0 & \\
 1 & 1 & \\
\end{array}
\end{align*}
Please implement each of the following Boolean expressions as a single static CMOS gate. **You can assume that inputs and their complements are available as inputs for the gate (just as in the diagram in part B).** Please carefully draw your nFETs and pFETs and label all inputs and outputs in your transistor diagrams.

(C) (4 points) \( F = (\overline{A} + \overline{B}) \cdot \overline{C} \)

(D) (4 points) \( F = (A + B) \cdot \overline{D} + C \)
Problem 4. Combinational Circuits (16 points)

(A) (4 points) Implement the function `lt2` that checks if `a` is less than `b` in Minispec. You may only use the bitwise logical operators: `~`, `&`, `|`, `^`. Assume that `a` and `b` are unsigned. When using negation (`~`), please include parentheses around the entire term being negated to clarify order of operation.

```verilog
function Bit#(1) lt2 (Bit#(2) a, Bit#(2) b);
    return ______________________________________________________
            ______________________________________________________;
endfunction
```
(B) (2 points) Fill out the truth table for the $lt_2$ function.

<table>
<thead>
<tr>
<th>$a[1]$</th>
<th>$a[0]$</th>
<th>$b[1]$</th>
<th>$b[0]$</th>
<th>$lt_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

(C) (3 points) Manually synthesize the $lt_2$ function into a combinational circuit using only inverters, and 2-input AND, OR, and XOR gates. Make sure to clearly label all inputs and outputs. For full credit, you should use at most 8 gates.
Consider the recursive Minispec function \( f_{\text{recursive}} \). Rewrite this function iteratively with a for loop. Your rewritten function should produce the same result but does not need to synthesize into the same circuit.

\[
\text{function Bit#(1) } f_{\text{recursive}}#(\text{Integer } n) \text{ (Bit#(n) } x) ;
\quad \text{if } (n == 1) \text{ return } x ;
\quad \text{else begin}
\quad \quad \text{Bit#(1) } bottom = f_{\text{recursive}}#(n/2)(x[n/2-1:0]) ;
\quad \quad \text{Bit#(1) } top = f_{\text{recursive}}#(n-n/2)(x[n-1:n/2]) ;
\quad \quad \text{return bottom} ^ \text{ top} ;
\quad \text{end}
\end{function}

\[
\text{function Bit#(1) } f_{\text{iterative}}#(\text{Integer } n) \text{ (Bit#(n) } x) ;
\end{function}
(E) (4 points) Consider the following Minispec function. Determine the result of running the function test(x) on each of the inputs specified below. Provide your result in \texttt{binary}.

\begin{verbatim}
function Bit#(4) test(Bit#(4) x) = case (x)
  2 : x | 1;
  3 : x + 3;
  5 : x << 2;
 10: x & 6;
 default: x ^ (x + 1);
endcase;
\end{verbatim}

Result of calling test(2) in binary (0b):______________

Result of calling test(5) in binary (0b):______________

Result of calling test(6) in binary (0b):______________

Result of calling test(10) in binary (0b):______________
Problem 5. Combinational and Sequential Logic Timing (19 points)

(A) (4 points) Given the timing parameters in the table below, what are the propagation and contamination delays of this circuit?

<table>
<thead>
<tr>
<th>Gate</th>
<th>$t_{pd}$ (ns)</th>
<th>$t_{cd}$ (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>XOR2</td>
<td>2</td>
<td>1.5</td>
</tr>
<tr>
<td>INV1</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td>AND2</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

Propagation delay (ns): __________

Contamination delay (ns): __________
(B) (4 points) Given the timing parameters in the table below, please indicate whether or not each constraint is satisfied by the circuit below and explain your answer. The registers are driven by a common clock with period \( t_{\text{clk}} = 10 \text{ns} \). Assume that IN satisfies the hold time constraint.

<table>
<thead>
<tr>
<th>Component</th>
<th>( t_{pd} ) (ns)</th>
<th>( t_{cd} ) (ns)</th>
<th>( t_{\text{SETUP}} ) (ns)</th>
<th>( t_{\text{HOLD}} ) (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register (R1, R2, R3)</td>
<td>2</td>
<td>1</td>
<td>1.5</td>
<td>4</td>
</tr>
<tr>
<td>Combinational Logic (CL1, CL2)</td>
<td>3</td>
<td>2</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Setup time constraint (circle one): Satisfied

Explanation:

Hold time constraint (circle one): Satisfied

Explanation:
(C) (4 points) We would like for this circuit to be able to function with a clock period of 9\text{ns}. Using the replacement parts below, modify the circuit so that it can operate using a clock period of 9\text{ns} while satisfying the setup and hold time constraints. For full credit, minimize the number of replacements made. To help you out, the circuit and original component timings are replicated above.

<table>
<thead>
<tr>
<th>Component</th>
<th>$t_{pd}$ (ns)</th>
<th>$t_{cd}$ (ns)</th>
<th>$t_{SETUP}$ (ns)</th>
<th>$t_{HOLD}$ (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register - Original</td>
<td>2</td>
<td>1</td>
<td>1.5</td>
<td>4</td>
</tr>
<tr>
<td>Combinational Logic - Original</td>
<td>3</td>
<td>2</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component</th>
<th>$t_{pd}$ (ns)</th>
<th>$t_{cd}$ (ns)</th>
<th>$t_{SETUP}$ (ns)</th>
<th>$t_{HOLD}$ (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register-New</td>
<td>1.5</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>CombinationalLogic-New</td>
<td>2</td>
<td>1</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**CL1:** REPLACED (with CombinationalLogic-New) NOT REPLACED  
**CL2:** REPLACED (with CombinationalLogic-New) NOT REPLACED  
**R1:** REPLACED (with Register-New) NOT REPLACED  
**R2:** REPLACED (with Register-New) NOT REPLACED  
**R3:** REPLACED (with Register-New) NOT REPLACED
(D) (3 points) Demonstrate that your updated circuit satisfies both the setup and hold time constraints.

(E) (4 points) After making the changes in part (C), what is the minimum clock period that will satisfy the setup and hold time constraints? What are the propagation and contamination delay of the updated circuit?

Minimum $t_{CLK}$ after replacements (ns): ________________

Propagation delay (ns): ________________

Contamination delay (ns): ________________
Problem 6. Finite State Machines (11 points)

Melon Usk bought his son Y Æ B-13 a brand new “Bop It”! The Bop It has five states, where each state corresponds to the Bop It asking the user to do some specific action: bop, pull, twist, flick, or spin. Y Æ B-13 thinks that the Bop It chooses its next state at random, but Melon actually hacked the toy so that he can choose the next state by inputting a zero or one into his Pear Watch.

<table>
<thead>
<tr>
<th>State</th>
<th>Input</th>
<th>Next State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bop</td>
<td>0</td>
<td>Spin</td>
</tr>
<tr>
<td>Bop</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Pull</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Pull</td>
<td>1</td>
<td>Flick</td>
</tr>
<tr>
<td>Twist</td>
<td>0</td>
<td>Pull</td>
</tr>
<tr>
<td>Twist</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Flick</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Flick</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Spin</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Spin</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

(A) (4 points) Using the provided partial information, complete both the truth table and the state-transition diagram. In the diagram, fill in the missing state labels and label each transition with either 0 or 1 to indicate which input value causes the transition.

Add missing information to diagram and table above

(B) (2 points) Given the following inputs, determine what the next state will be (bop, pull, twist, flick, or spin) after the last input is processed. Assume the FSM starts at the Bop state for each series of inputs. Assume that an input of 100, for example, means the FSM got a 1 followed by two 0s.

(i) 0101

(ii) 111

(iii) 101

Final state: ________________

Final state: ________________

Final state: ________________
(C) (1 points) How many flip flops does the Bop It FSM require to encode all possible states?

Number of Flip Flops Required: __________________

(D) (2 points) Melon’s Pear Watch is glitching and he can now only input ones. After at least two more state transitions, what states of the Bop It will Melon no longer be able to access?

Which State(s) Become Inaccessible? ________________________________

(E) (2 points) Part E refers to a new FSM that is completely separate from the FSM in parts A-D. Below is a state-transition diagram for a state machine with possible inputs 0 and 1. Is the state-transition diagram valid? If yes, how many flip flops are needed to encode all possible states? If no, explain why not.

Is the state-transition diagram valid? (circle one):  VALID  INVALID

If VALID:

Number of Flip Flops Needed: _____________

If INVALID:

Explanation:

END OF QUIZ 1!