Q1.a.  
<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>(AB)</th>
<th>(A' C)</th>
<th>((AB) + (A'C))'</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</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>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</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>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Q1.b.  
A -+ B
H
C

Q2.a.  
<table>
<thead>
<tr>
<th>A</th>
<th>S</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Q2.b.  
A -+ S
X
A -+ S
Y

Q3.a.  
<table>
<thead>
<tr>
<th>P</th>
<th>Q</th>
<th>P&gt;Q</th>
<th>P==Q</th>
<th>P&lt;Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</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>
</tbody>
</table>

Q3.b.  

Q4.a.  
assign X = (A B B) | (~A B C) | (B B C);  
0.5 total if no semicolons

Q4.b.  
assign {X, Y} = {A B ~S, A B S};

Q5.a.  
module DEMUX21 (input A, input S, output X, output Y);
assign X = A B ~S;
assign Y = A B S;
endmodule

Q5.b.  
module DEMUX41 (input A, input [1:0] S, output [3:0] Z);
DEMUX21 dm1 (A, ~S[1] & S[0], Z[0], Z[1]);
DEMUX21 dm2 (A, S[1] & S[0], Z[2], Z[3]);
endmodule

Q6.  
output reg X
output reg Y
always C begins
X = A B ~S;
Y = A B S;
end
Q7. 4 pts, 1 per clk, .5 → Q, .5 → Q

(Q8-10: see last page)

Q11. 5 pts

D

Clk

Setup: input stable before clock edge 1 pt
Hold: input stable after clock edge 1 pt

Q12. 10 pts

a.
1 pt: A buffer that can turn its output "off". States are High, Low, and High-Z. Hi-Z is high impedance, that is the output is functionally disconnected.
2 pt: A drives high, others Hi-Z
3 pt: Drive same value (High or Low) nothing. Drive high and low ⇒ short.
4 pt: Undefined. Hi-Z is floating.

b.
6 pts
2 pts: A BJT where

The pull-up causes the collector to be normally high.
1 pt: A does nothing to drive 1. Other devices also do nothing.
1 pt: A turns on the BJT, grounding the circuit.
1 pt: Insert the whole setup (the high & use a pull down resistor)
1 pt: The size of the resistor, the capacitance of the collector network.

Rubric

Q1: 2 pts
Q2: 2 pts
Q3: 2 pts
Q4: 2 pts
Q5: 6 pts
Q6: 2 pts
Q7: 4 pts
Q8: 5 pts
Q9: 5 pts
Q10: 5 pts
Q11: 1 pt
Q12: 1 pt

Total: 10 pts
Q8. Build a module-4 counter: \( 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \)

<table>
<thead>
<tr>
<th>PS</th>
<th>NS</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</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>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\[ D_1 = Q_1 \oplus Q_0 = Q_1 \cdot Q_0 + Q_1 \cdot Q_0' \]
\[ D_0 = Q_0' \]

Q9 - 5 pts

a. 3 bits are used.
   Could have been done in 2. 1pt

b. always C4
   \[ \text{pro} \]
   * clear, readable for complex state machines
   * compiler can verify all cases covered
   \[ \text{con} \]
   * can be verbose
   * easy to accidentally miss a case

assigns
   \[ \text{pro} \]
   * concise, easy to read for simple expressions
   \[ \text{con} \]
   * hard to understand w/more states
   * hard to spot typos

1pt per claim, up to 3