1. (10 points) Modify the following VHDL model to use process(es) instead of blocks.

```vhdl
use WORK.TSL.all, WORK.SYS.all;
entity I8212 is
  generic (GDEL, FFDEL, BUFDEL: TIME);
  port (DI: in WORD;
   DO: out WORD;
   NDS1, DS2, MD, STB, NCLR: in BIT;
   NINT: out BIT := '1');
end I8212;
architecture BEHAVIOR of I8212 is
begin
  I8212_BLK: block
  signal S0, S1, S2, S3, S4: BIT;
  signal SRQ: BIT;
  signal Q: WORD;
  begin
    INT_BLK:
    block (S1='1' and S4='1')
    begin
      Q <= guarded DI after FFDEL;
      Q <= "00000000" after FFDEL when (S1='0' and S4='0') else Q;
      DO <= Q after BUFDEL when (S3='1') else
          "ZZZZZZZZ" after BUFDEL;
    end block INT_BLK;
    S0 <= not NDS1 and DS2 after GDEL;
    S1 <= (S0 and MD) or (STB and not MD) after (2*GDEL);
    S2 <= (S0 nor (not S4)) after GDEL;
    S3 <= (S0 or MD) after GDEL;
    S4 <= (S1 OR NCLR) after GDEL;
    SRQ <= '1' after FFDEL when (S2='0') else
       '0' after FFDEL when (S2='1') and (not STB'STABLE) and (STB='0') else
       SRQ;
    NINT <= not SRQ nor S0 after GDEL;
  end block I8212_BLK;
end BEHAVIOR;
```
2. (5 points) What kind of hardware element will be inferred by a synthesis tool from the following model?

```vhdl
library ieee;
use ieee.std_logic_1164.all;

entity WIDGET is
    Port (A, B : in SIGNED (0 to 2);
           CLK, RESET : in std_logic;
           Z : out SIGNED(0 to 2));
end WIDGET;

architecture EXAMPLE of WIDGET is
begin
    process (CLK, RESET)
    begin
        if (RESET = '1' then)
            Z <= '0';
        elsif (CLK = '1') then
            Z <= A nor B;
        end if;
    end process;
end EXAMPLE;
```

3. (5 points) Draw the transistor-level diagram of a CMOS inverter.
Consider the following VHDL code:

--- Entity declaration
---

entity SCHED2 is
  port (A, B, C, D, E, F: in INTEGER;
        CLK : in BIT;
        W, X, Y: out INTEGER);
end SCHED2;

--- Architecture declaration
---

architecture HIGH_LEVEL of SCHED2 is
  signal Z: INTEGER;
begin
  X <= (A - B) * Z;
  Y <= (A * B) + Z;
  Z <= (C * D) + D * (E + F);
end HIGH_LEVEL;

4. (15 points) The following tasks refer to the VHDL code above. Assume that there are no hardware constraints.
   a. (5 points) Draw a data flow graph.
b. (5 points) Derive an ASAP schedule.
c. (5 points) Derive an ALAP schedule.
5. (10 points) Derive a list schedule using the critical path priority metric for the VHDL code above, using the following hardware constraint; all operations are done in an ALU module and there are two ALU modules available.

6. (10 points) Write a VHDL entity and architecture of a two-input AND gate with the generics, TPLH and TPHL, which reflect the time for the output to make a low to high or high to low transition, respectively.
7. (10 points) For the data lifetime chart shown, use the left edge algorithm to obtain an efficient register allocation.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S3</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S4</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S5</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8. (1 point) A(n) __________________ is an integrated circuit produced for a specific application and produced in relatively small volumes.

9. (1 point) __________________ is an annoyingly strongly typed language.

10. (1 point) ________________________ removes the hierarchy in a circuit, so the circuit has a single level.

11. (1 point) _________________ is the process of transforming a behavioral description into a structural gate-level circuit.
12. (15 points) Design a Moore state machine in VHDL which converts NRZ (non-return-to-zero) coding to Manchester coding. In NRZ coding, each bit is transmitted for one bit time without any change. For the Manchester code, a 0 is transmitted as 0 for the first half of the bit time and 1 for the second half, but a 1 is transmitted as 1 for the first half and 0 for the second half. In order to do this conversion, use a clock(CLOCK2) that is twice the frequency of the basic clock. Note that if the NRZ bit is 0, it will be 0 for two CLOCK2 periods. Similarly, if the NRZ bit is 1, it will be 1 for two CLOCK2 periods. Your design should have an active low synchronous reset and work with CLOCK2.

<table>
<thead>
<tr>
<th>NRZ</th>
<th>0 0 1 1 0 0 1 1 0 0 0 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOCK2</td>
<td></td>
</tr>
<tr>
<td>Manchester</td>
<td>0 1 1 0 0 1 1 0 0 1 0</td>
</tr>
</tbody>
</table>

![Diagram of NRZ and Manchester coding conversion](image-url)
13. (10 points) Develop a synthesizable VHDL entity and architecture for an M-N flip-flop with an active high asynchronous reset. An M-N flip-flop works as follows:
   - If MN = 00, the next state of the flip-flop is 0.
   - If MN = 01, the next state of the flip-flop is the same as the present state.
   - If MN = 10, the next state of the flip-flop is the complement of the present state.
   - If MN = 11, the next state of the flip-flop is 1.

14. (5 points) If the NRE costs for FPGA and CBIC circuits are $21,000 and $187,000, respectively, and the cost of individual parts for FPGA and CBIC circuits are $18 and $5, respectively, what is the break-even manufacturing volume for these two types of circuits?

15. (1 point) _________________________ is the hardest problem.