This exam is open book and open notes. You have 50 minutes. Credit for problems requiring calculation will be given only if you show your work.

1. Floating Point  

Translate the IEEE single-precision floating point number shown below to its decimal equivalent and write it in the space provided below.

```
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Value of number: ________________
2. Single-Cycle Processor Design 25 Points

Modify the single-cycle datapath and control to implement a new R-format instruction “\texttt{lwx}” (load word - indexed). It will be used in assembly language as follows:

\texttt{lwx \ rd, \ rs(rt)}

The register transfer for this instruction is as follows:

\[ \text{Reg}[\text{rd}] \leftarrow \text{MEM}[\text{Reg}[\text{rs}] + \text{Reg}[\text{rt}]]; \]

The \texttt{lwx} instruction is to be encoded as an R-Type instruction, with \texttt{opcode=0} and \texttt{funct=48}. Mark the necessary changes to the datapath and control on the diagrams shown below. If no changes are necessary, write this down on your paper.

\begin{center}
\textbf{Control Table}
\end{center}

\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|}
\hline
\textbf{Instr} & \textbf{RegDst} & \textbf{ALUSrc} & \textbf{MemitReg} & \textbf{RegWrite} & \textbf{MemRead} & \textbf{MemWrite} & \textbf{Branch} & \textbf{ALUOp1} & \textbf{ALUOp0} & \textbf{Jump} \\
\hline
\texttt{lwx} & & & & & & & & & & \\
\hline
\end{tabular}
3. **Multicycle Processor Timing**

   This problem refers to the multicycle datapath shown on the next page. Assume that the multicycle datapath components have the same delay characteristics as the single-cycle components described on page 373 of the book – ALU and memory have a 2ns delay, register file read and write each have a 1ns delay.

   (a) Assume that all other components have no delay. What is the minimum clock period at which this design can operate properly?

   (b) Now assume that in addition to the delays given above, the delay of *multiplexers* is 0.1ns. What is the minimum clock period at which this design can operate properly?

4. **Multicycle Processor Design**

   We wish to modify the multicycle processor design to implement the $\text{lwx}$ instruction described in Problem 2. Modify the attached datapath and state diagram to show any necessary changes to this design to accomplish this.
Insert multicycle datapath here
5. **Verilog**

This problem refers to the Verilog code shown on the next page

(a) Draw a diagram on the next page which shows the module instances and connections created in the module "exam code.

(b) Is the reset input of the “dff” module an asynchronous or synchronous reset? Why?
module fulladder(a, b, cin, sum, cout);
    input a, b, cin;
    output sum, cout;

    assign sum = a ^ b ^ cin;
    assign cout = a & b | a & cin | b & cin;
endmodule

module dff(clk, reset, d, q);
    input clk, reset, d;
    output q;
    reg q;

    always @(posedge clk)
        if (reset) q <= 1'b0;
        else q <= d;
endmodule

module exam(clk, reset, in_a, in_b, out);
    input clk, reset, in_a, in_b;
    output out;

    wire ff_in, ff_out;

    fulladder FA(in_a, in_b, ff_out, out, ff_in);

    dff FF(clk, reset, ff_in, ff_out);
endmodule