1. [15%] Design the circuit of a 2-bit magnitude comparator shown below.

\[ A \{ a_0, a_1 \} \rightarrow A > B, A = B, A < B \]
\[ B \{ b_0, b_1 \} \]

\[ a_1 a_0 > b_1 b_0 \Rightarrow (A > B) = 1, (A = B) = 0, (A < B) = 0 \]
\[ a_1 a_0 = b_1 b_0 \Rightarrow (A > B) = 0, (A = B) = 1, (A < B) = 0 \]
\[ a_1 a_0 < b_1 b_0 \Rightarrow (A > B) = 0, (A = B) = 0, (A < B) = 1 \]

(a) [3%] Derive the truth table in term of inputs.
(b) [3%] Express the outputs in sum of minterms.
(c) [3%] Simplify the output functions in sum of products.
(d) [3%] Draw the logic diagram with AND, OR and NOT gates.
(e) [3%] Draw the logic diagram with NAND gates only.

2. [15%] A sequential circuit has one input, \( x \). The state diagram is shown below.

\[ \begin{array}{c}
00 & 01 & 10 & 11 \\
1 & 0 & 1 & 0
\end{array} \]

(a) [3%] Obtain the state table.
(b) [3%] Reduce the number of states and redraw the state diagram.
3. [10%] Consider two processors P1 and P2, of the same instruction set. There are three classes of instructions (A, B, and C) in the instruction set. Processor P1 has a clock rate of 500 MHz, and processor P2 has a clock rate of 400 MHz. Two different compilers C1 and C2 can run on processor P1 as well as on processor P2. Assume that each compiler uses the same number of instructions for a given test program but that the instruction mix is as described in the following table. The table also contains the average number of cycles for each instruction class (CPI) on P1 and P2.

<table>
<thead>
<tr>
<th>Instruction class</th>
<th>CPI for P1</th>
<th>CPI for P2</th>
<th>C1 usage</th>
<th>C2 usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>2</td>
<td>30%</td>
<td>50%</td>
</tr>
<tr>
<td>B</td>
<td>3</td>
<td>2</td>
<td>50%</td>
<td>25%</td>
</tr>
<tr>
<td>C</td>
<td>4</td>
<td>3</td>
<td>20%</td>
<td>25%</td>
</tr>
</tbody>
</table>

(a) [2%] If you purchase processor P2, what is the average CPI for each compiler?
(b) [3%] If you purchase processor P2, which compiler will execute faster according to execution time?
(c) [3%] Using compiler C1 on both P1 and P2, which processor is faster?
(d) [2%] Using compiler C1 on both P1 and P2, what are the MIPS (million instructions per second) ratings for each processor?

4. [10%] For the following graphical representation of a control unit, the task is to build a combinational logic to generate outputs and a state feedback register to hold the current state, as shown in the block diagram. The outputs of the combinational logic are the next-state number and the control signals to be asserted for the current state. The control signals (A, B, C, D, E, and F) are activated when 1 is specified and deactivated otherwise. Draw an implementation of the combinational logic using basic gates. You may use a programmed logic array (PLA) to simply your drawing.
5. [10%] The following figure shows the datapath with the control unit and the control signals of a single-cycle processor. The processor supports only three instruction formats as shown below. The available control signals have been classified into five classes (A, B, C, D, and E), assuming 1 when activated, 0 when deactivated, and X for "don't care". The following table defines how the control signals should be set for each class; however, the table only provides partial information. For each of the following instructions, specify its
corresponding class (A to E) or designate X if none of the known classes is appropriate.

(a) add $s1, $s2, $s3  # $s1 = $s2 + $s3
(b) beq $s3, $s4, 25   # if ($s3 == $s4) go to PC+4+100
(c) sw $s1, 100($s3)   # Memory[$s3+100] = $s1
(d) addi $t1, $t2, 20  # $t1 = $t2 + 20
(e) lw $t1, 0($s2)     # $t1 = Memory[$s2+0]
(f) slt $s1, $s2, $s3  # if($s2 < $s3) $s1 = 1 else $s1 = 0
(g) bne $s1, $s2, 25   # if ($s1 != $s2) go to PC+4+100
(h) slti $s2, $s3, 100  # if($s3 < 100) $s2 = 1 else $s2 = 0
(i) j 2500             # go to 10000
(j) sub $s1, $zero, $s1 # $s1 = $zero - $s1
6. [20%] For the following 5-stage pipelined datapath, the operation times for the major functional units in this datapath are 200 ps for memory access, 200 ps for ALU operation and 100 ps for register file read or write. Please answer the following questions for executing the following codes:

\[
\begin{align*}
&\text{l} \ w \ $1, \ 100(\$2) \\
&\text{sub} \ $3, \ $1, \ $5 \\
&\text{add} \ $2, \ $5, \ $6
\end{align*}
\]

(a) [2%] How long does it take to execute this program code on a nonpipelined processor? And on the above 5-stage pipelined datapath?

(b) [4%] How long does it take to execute this code, both in ps and in clock cycles?

(c) [4%] At the end of the fifth cycle of execution, which registers are being read? Which registers are being written?

(d) If we have a program of \(10^4\) instructions in the sequence of “\(\text{l} \ w, \ \text{add}, \ \text{l} \ w, \ \text{add}, \ ..\)” The add instruction depends (and only depends) on the \(\text{l} \ w\) instruction right before it. The \(\text{l} \ w\) instruction also depends (and only depends) on the add instruction right before it. If the program is executed on the above pipelined datapath:

i. [5%] With forwarding, what would be the actual CPI?

ii. [5%] Without forwarding, what would be the actual CPI?
7. [10%] The memory-stall clock cycles come primarily from cache misses. Assume an instruction cache miss rate for a program is 2% and a data cache miss rate is 4%. The frequency of all loads and stores is 36%. Please answer the following questions to get the impact of cache performance on processor performance.

(a) [5%] If a processor has a CPI of 1.0 without any memory stalls and the miss penalty is 100 cycles for all misses, determine how much faster the processor would run with a prefect cache that never missed?

(b) [5%] Suppose we increase the performance of the processor by doubling its clock rate. Since the main memory speed is unlikely to change, assume that the absolute time to handle a cache miss does not change. How much faster will the processor be with the faster clock, assuming the same miss rate?

8. [10%] Assume a memory system that supports interleaving either 4 reads or 4 writes, as shown in the following figure. Given the following memory addresses in order as they appear on the memory bus: 3, 13, 21, 2, 51, 37, 9, 16, 8, 41, which ones will result in a bank conflict?