Problem 1
Design a sequential circuit to detect the sequence 11011. The sequences may overlap. Use a PLA to implement the combinational logic and D flip flops to implement the memory elements.

Problem 2
Construct a 3-bit (modulo 8) counter using three D flip-flops and a selection of gates. The inputs consist of a reset signal that resets the counter to 0 and an inc signal that increments the counter. The outputs are the value of the counter. When the counter has a value 7 and is incremented, it wraps around to a value 0.

Problem 3
Using three D flip-flops and a PLA implement a 3-bit Gray code counter. The inputs consist of a reset signal that resets the counter to 0 and an inc signal that increments the counter. The outputs are the value of the counter. A Gray code is a sequence of binary numbers with the property that no more than 1 bit changes in going from one element of the sequence to another. For example, a 3-bit binary Gray code is given by: 000, 001, 011, 010, 110, 111, 101, 100. The Gray code is cyclic so that the value after 100 in the sequence is 000.

Problem 4
There are three 2-bit registers X, Y and Z. Each bit in the register is a clocked D flip-flop. Design the logic to perform the following operations:

AND: $Z \leftarrow X \land Y$
OR: $Z \leftarrow X \lor Y$
NOP: $Z \leftarrow Z$
CLR: $Z \leftarrow 0$

Here $\land$ and $\lor$ denote the bit-wise AND and bit-wise OR of the corresponding registers respectively. Use appropriate multiplexers in your design. The appropriate operation AND, OR, NOP or CLR is chosen using a 2-bit control word C such that $C = 0$ represents AND; $C = 1$ represents OR, $C = 2$ represents NOP and $C = 3$ represents CLR. Show neatly the resulting logic diagram. What condition do the D flip-flops need to satisfy in order for the design to work?

Problem 5
A 256 word memory has its words numbered from $(0)_{10}$ to $(255)_{10}$. Define the address bits for each of the following. Each address bit should be specified as a 0, 1 or d (don't care).
(a) Word $(48)_{10}$
(b) Lower half of the memory, i.e., words $(0)_{10}$ through $(127)_{10}$
(c) Upper half of the memory, i.e., words $(128)_{10}$ through $(255)_{10}$
(d) Even memory words i.e., $(0)_{10}, (2)_{10}, (4)_{10}, \ldots, (254)_{10}$
(e) Any of the eight words $(48)_{10}$ through $(55)_{10}$.

**Problem 6**
Design a $16K \times 8$ memory using
(a) only $1K \times 1$ memory modules.
(b) only $2K \times 4$ memory modules.
(c) only $1K \times 8$ memory modules.
Each module is capable of address decoding, has tristate outputs (0, 1 and not connected (nc)), and read enable, write enable and chip enable inputs. Draw a clear logic circuit.

**Problem 7**
Show how a $16K \times 32$ RAM module can be converted into a $64K \times 8$ RAM. Treat the $16K \times 32$ RAM module as a single unit that cannot be altered internally, i.e., only introduce logic elements that are external to the RAM module. Draw a clear logic circuit. Assume that the $16K \times 32$ RAM module is capable of address decoding, has tristate outputs (0, 1 and not connected (nc)), and read enable, write enable and chip enable inputs. Implement only the memory read portion of the circuit.