1.  
   (1) (5%) Add the following numbers in binary using 2’s complement to represent negative numbers. Use a word length of 6 bits (including sign) and indicate if an overflow occurs.  
       (a) 13 + (-12)  
       (b) (-11) + (-21)  
   (2) (5%) Devise a scheme for converting base 2 numbers directly to base $2^n$ ($2 \leq n \leq 4, n \in \mathbb{N}$) numbers. Use your method to convert the following number to base 16: (11100011101101101111)2  
   (3) (5%) Prove that: $(X' + Y')(X \equiv Z) + (X + Y)(X \oplus Z) = (X \oplus Y) + Z'$  
   (4) (5%) Use a 4-variable Karnaugh map with the method of map-entered variables to find a minimum sum-of-products expression for function $Z$:  
       $Z(A, B, C, D, E, F, G) = \sum m(0, 3, 13, 15) + \sum d(1, 2, 7, 9, 14) + E(m_0 + m_1) + F'm_2 + G'm_3$  
   (5) (5%) The elevator door should open if the elevator is stopped, it is level with the floor, and the timer has not expired, or if the elevator is stopped, it is level with the floor, and the “open” button is pressed. Write an equation for the variable $Z$ which is 1 iff the door should open. Express $Z$ as a minterm expansion. (Use m-notation.)

2. A switching circuit has two control inputs ($C_1$ and $C_2$), two data input ($X_1$ and $X_2$) and one output ($Z$). The circuit performs logic operations on the two data inputs, as shown in the following table:  
   (1) (5%) Derive a truth table for $Z$.  
   (2) (5%) Use a Karnaugh map to find the minimum product-of-sums form of the circuit.

<table>
<thead>
<tr>
<th>$C_1$</th>
<th>$C_2$</th>
<th>Logic Function Performed by Circuit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>$X_1X_2$</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>$X_1 \oplus X_2$</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>$X'_1 + X_2$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>$X'_1 \equiv X_2$</td>
</tr>
</tbody>
</table>

3.  
   (1) (8%) List all of the prime implicants, find the essential prime implicants, and find the minimum sum-of-products expression using a K-map for the function:  
       $F(a, b, c, d, e) = \sum m(0, 1, 4, 5, 9, 10, 11, 12, 14, 18, 20, 21, 22, 25, 26, 28)$  
   (2) (7%) Find all prime implicants of the following function using the Quine-McCluskey method; and then find all minimum sum-of-products solutions using Petrick’s method:  
       $F(A, B, C, D) = \sum m(9, 12, 13, 15) + \sum d(1, 4, 5, 7, 8, 11, 14)$
4. Design a 3-bit counter that is initially reset to the 000 state and is able to count in the order of 000 → 110 → 101 → 010 → 001 → 111 → 000 as well as in its reverse order as shown in the following figure.

```
  000
  ↓   ↓
  111  110
  ↓   ↓
  001  101
  ↓   ↓
  010
```

Design this counter.

(1) (5%) Design with T flip-flops for the counter to count clockwise. List the next-state tables and derive the corresponding input equations.

(2) (5%) Design with J-K flip-flops for the counter to count counter-clockwise. List the next-state tables and derive the corresponding input equations.

(3) (5%) Assume an input signal UD is used to control the counting direction of the counter. Design the up-down counter with D flip-flops for the counter to count counter-clockwise when UD=0 and clockwise when UD=1. Draw the corresponding logic circuit diagram.
5. For the state diagram of a mealy machine shown below

![State Diagram](image)

where the x and y are the inputs and z is the output of the state machine:

1. (5%) Draw the state transition table of the state diagram.
2. (5%) Design a synchronous circuit with D flip-flops that realizes this state diagram with the corresponding state assignment of S₀=00, S₁=01, S₂=11 and S₃=10. The circuit is reset to state S₀ and is activated at the rising-edge of the clock signal CLK.
3. (5%) Redesign a circuit that realizes the state table with the same state assignment using S-R flip-flops.

6. Design a circuit that can perform filtering using a first order FIR filter of the form: y(n) = a₀ u(n) + a₁ u(n-1). Assume that the input u(n), ∀ n, and the filter coefficients a₀ and a₁ are all positive integers and 8-bit in width. In addition, only one multiplier and one adder are allowed to realize the circuit.

1. (5%) Design the data path for your circuit which consists of one multiplier, one adder and the necessary registers and multiplexers. Note that you need to mark the corresponding bit-width for each data bus.
2. (5%) Draw the timing diagram for all of the control signals and the switching of data buses by referencing to the rising-edge of a clock signal.
3. (5%) Design a state machine that generates the control signals of the data-path using the state diagram.
4. (5%) Realize this state machine with D flip-flops.