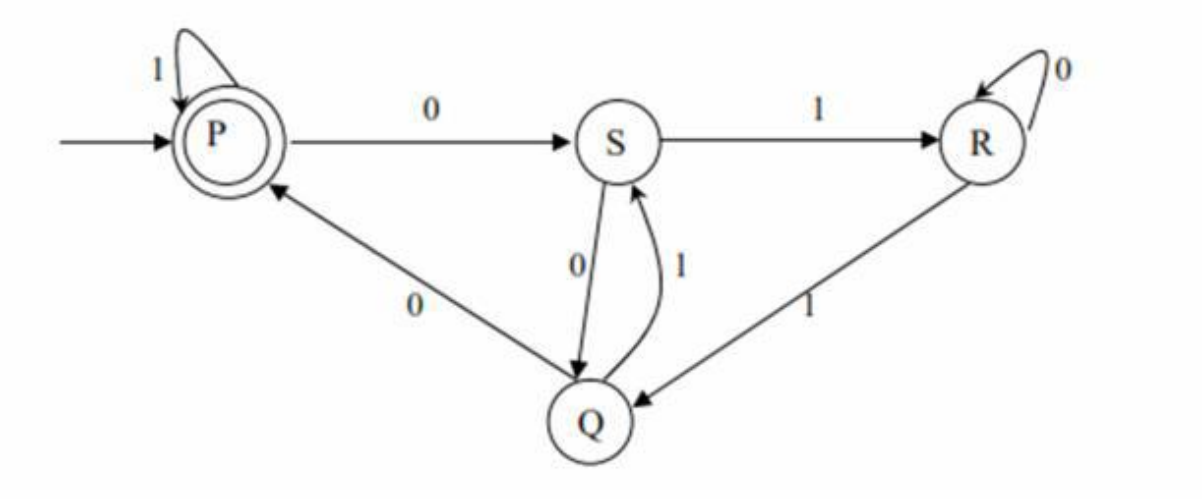
Tutorial Practice Set 3
Automata and Compiler Design

Q1. Design Moore machine that takes an input bit string *b* and produces the output NOT(*b*).

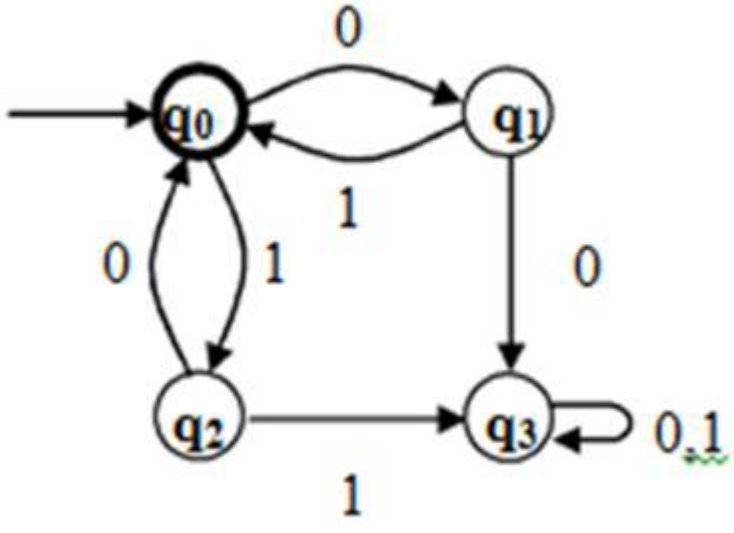
Q2. Design Moore machine that halves a binary number, truncating any decimal places.

Q3. Convert Q1 and Q2 Moore machine into Mealy machine.

Q4. Convert the following DFA to Regular Expression



Q6. Obtain a regular expression for the FA shown below**:**



Q5. Obtain an NFA to accept the following language L = {w | w =$(ab)^n$ or $(aba)^n$ where n >= 0}

Q6.

Give regular expressions that generate each of the following languages. In all cases, the alphabet is $\Sigma = \{a, b\}$.

(a) The language $\{ w \in \Sigma^* \mid |w| \text{ is odd} \}$.

(b) The language $\{ w \in \Sigma^* \mid w \text{ has an odd number of } a\text{'s} \}$.

(c) The language $\{ w \mid w \text{ contains at least two } a\text{'s, or exactly two } b\text{'s} \}$.

(d) The language $\{ w \in \Sigma^* \mid w \text{ ends in a double letter} \}$. (A string contains a *double letter* if it contains aa or bb as a substring.)

(e) The language $\{ w \in \Sigma^* \mid w \text{ does not end in a double letter} \}$.

(f) The language $\{ w \in \Sigma^* \mid w \text{ contains exactly one double letter} \}$. For example, *baaba* has exactly one double letter, but *baaaba* has two double letters.

Q7. Convert the regular expression $(((00)^*(11))\cup 01)^*$ into an NFA.

Q8.

Prove that the following languages are not regular.

(a) $A_1 = \{ www \mid w \in \{a, b\}^* \}$.

(b) $A_2 = \{ w \in \{a, b\}^* \mid w = w^{\mathcal{R}} \}$.

(c) $A_3 = \{ a^{2n} b^{3n} a^n \mid n \geq 0 \}$.

(d) $A_4 = \{ w \in \{a, b\}^* \mid w \text{ has more } a\text{'s than } b\text{'s} \}$.