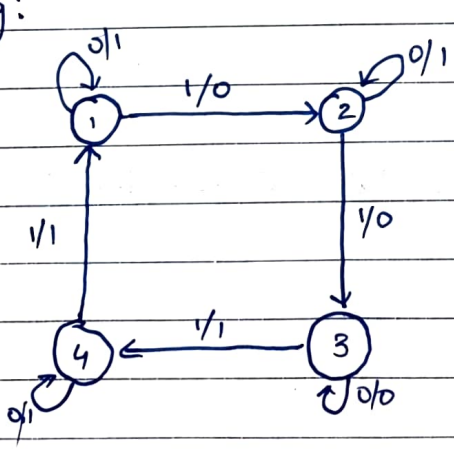
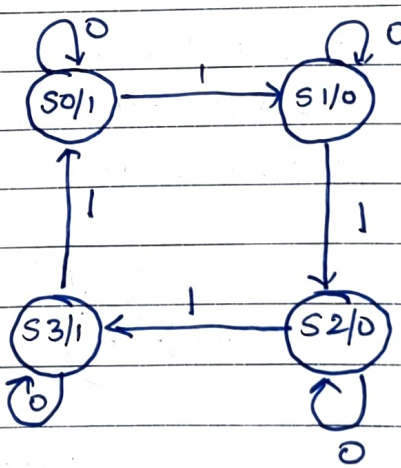
FA
TUTORIAL 3

Junaid. Girkar
B00041900577

Q1 Differentiate between Mealy and Moore machine

MEALY MACHINE	MOORE MACHINE
• output depends upon present state as well as present input.	• output depends only upon present state
• If input changes, output also changes.	• If input changes, output does not change
• less number of states are required	• More number of states are required
• There is more hardware required for circuit implementation.	• There is less hardware requirement for circuit implementation.
• They react faster to inputs	• They react one clock cycle later to inputs
• Asynchronous output generation	• Synchronous output and state generation
• output is placed on transitions.	• output is placed on states
• E.g: 	• E.g: 

Q2 Design a mealy machine to find 2's complement of given binary number. Convert into moore machine

ANS 2's complement : It is the mathematical operation on binary numbers. It is used for computation as a method of signed number representation. Its complement with respect to 2^N defines the two's complement an N-bit number.

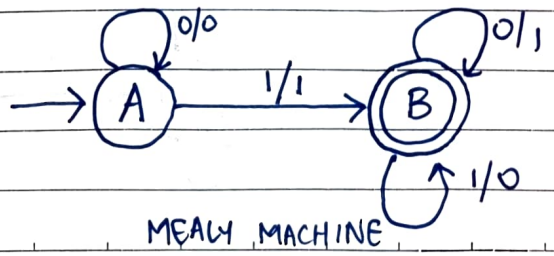
LOGIC : First calculate 1's compliment of binary number, convert 1 to 0 and 0 to 1 and then add 1 to it.

DESIGN A MEALY MACHINE :

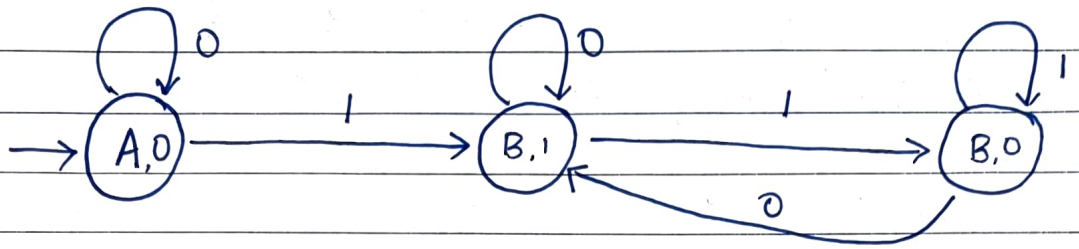
- Take initial state A
- If there are n number of zeros at initial state, it will remain at initial state.
- Whenever first input 1 is found then it gives output 1 and go to state B.
- In state B, if input is zero, output will be 1. And if input is 1 then output will be 0
- And then set state B as final state

THE APPROACH

- Start from right to left
- Ignore all 0's
- When 1 comes, ignore it and then take 1's compliment of every digit



MEALY MACHINE



MOORE MACHINE

Q3 Minimize the following DFA using equivalence theorem.

	Q \ Σ	0	1
→	q_0	q_1	q_5
	q_1	q_6	* q_2
*	q_2	q_0	* q_2
	q_3	q_2	q_6
	q_4	q_7	q_5
	q_5	q_2	q_6
	q_6	q_6	q_4
	q_7	q_6	* q_2

STEP 1:

π_0 : $\{q_2\}$ $\{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$

STEP 2:

$\delta(q_0, 0) = q_1$

$\delta(q_1, 0) = q_6$

$\therefore q_0 q_1$ not possible

$\delta(q_0, 1) = q_5$ ✓

$\delta(q_1, 1) = q_2$ ×

Using Equivalence theorem,

$\pi_0 = \{q_0, q_1, q_4, q_5, q_6, q_7\}\{q_2\}$

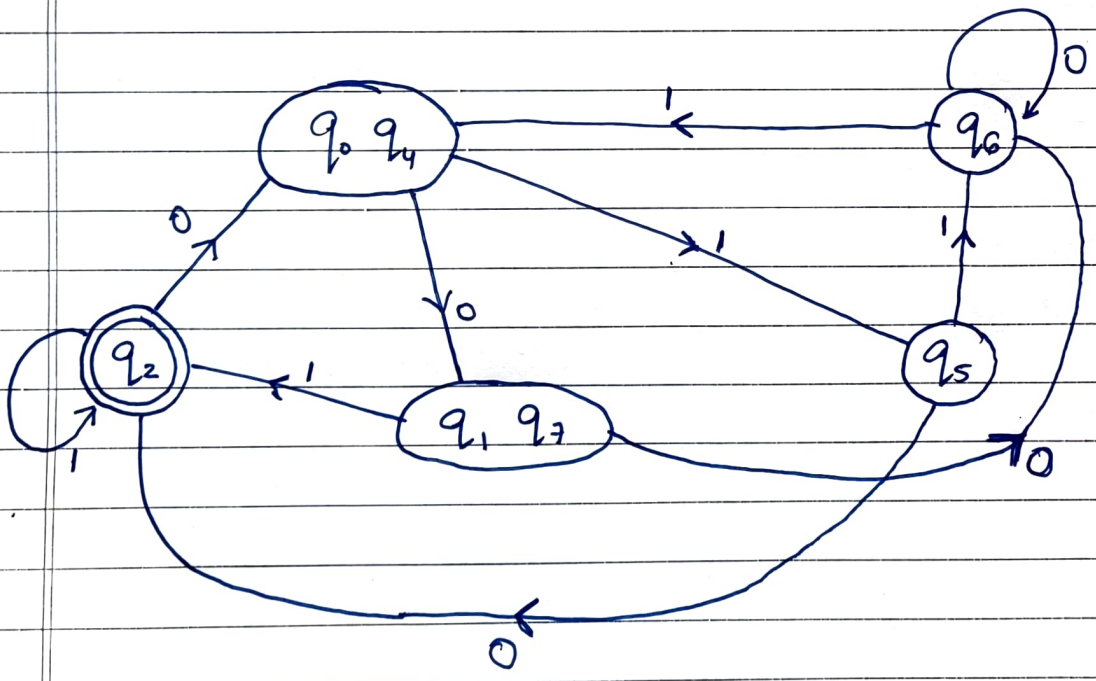
$\pi_1 = \{q_0, q_4, q_6\}\ \{q_1, q_7\}\ \{q_5\}\ \{q_2\}$

$\pi_2 = \{q_0, q_4\}\ \{q_6\}\ \{q_1, q_7\}\ \{q_5\}\ \{q_2\}$

$\pi_3 = \{q_0, q_4\}\ \{q_6\}\ \{q_1, q_7\}\ \{q_5\}\ \{q_2\}$

$\because \pi_2 = \pi_3$

$\therefore$ Minimized DFA is



Q4 List various applications of Finite Automata. Explain lexical analyzer.

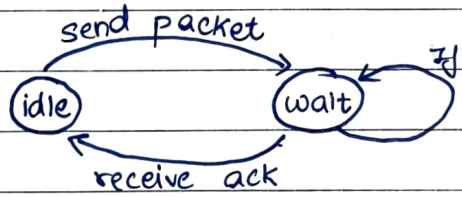
ANS Applications of finite automata include string matching algorithms, network protocols and lexical analyzers.

1) STRING PROCESSING :

Consider finding all occurances of a short pattern string within a long text string. This can be done by processing the text through a DFA : the DFA for all strings that end with the pattern string. Each time the accept state is reached, the current position of text is output.

2) FINITE STATE MACHINE :

A finite state machine is an FA together with actions on the arcs. A trivial example for a communication link :-



3) STATECHARTS :

Statecharts model tasks as a set of states and actions. They extend FA diagrams.

Q4 list vario

4) LEXICAL ANALYSIS:

In compiling a program, the first step is lexical analysis. This isolates keywords, identifiers, etc. while eliminating irrelevent symbols. A token is a category, for example "identifier", "relational operator" or specific keyword

For Example:

token	RE
Keyword then	then
variable name	[a-zA-Z][a-zA-Z0-9]*

5) LEXICAL ANALYZER:

A lexical analyzer takes source code as a string and outputs sequence of tokens.

For example,

```
for i = 1 to max do
    x[i] = 0 ;
```

might have token sequence

for id = num to id do id [id] = num sep

As a token is identified, there may be an action. For example, when a number is identified, its value is calculated.