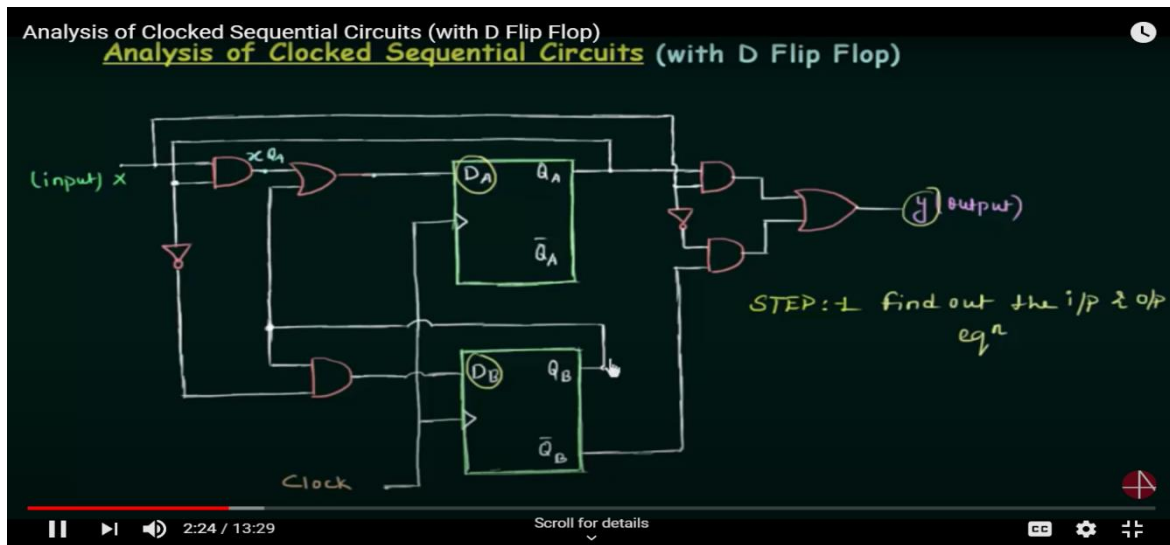


DAILY ASSESSMENT FORMAT

| | | | |
|--------------------|---|---------------------|-------------------|
| Date: | 29-05-2020 | Name: | Sushmitha R Naik |
| Course: | Logic Design | USN: | 4AL17EC090 |
| Topic: | 1. Analysis of clocked sequential circuits 2. Digital clock design | Semester & Section: | 6 th B |
| GitHub Repository: | Sushmitha_naik | | |

FORENOON SESSION DETAILS

Image of session



Analysis of Clocked Sequential Circuits (with D Flip Flop)

| Q_A | Q_B | x | Q_A | Q_B | y |
|-------|-------|-----|-------|-------|-----|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 |

STEP:-3 state diagram

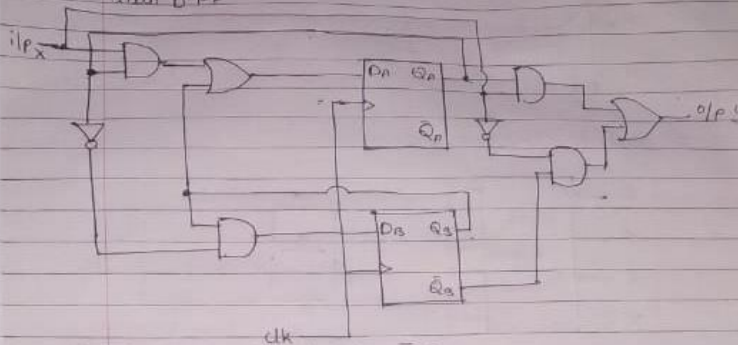
$S_0 = Q_A Q_B = 00$
 $S_1 = 01$
 $S_2 = 10$
 $S_3 = 11$

$y = 1 \cdot 1 + 0 \cdot 0 = 1$
 $y = 0 \cdot 1 + 1 \cdot 0 = 0$

Report – Report can be typed or hand written for up to two pages.

Day 2

Analysis of clocked Sequential Circuits with D FF



$$D_1 = x \oplus Q_1 + Q_2 \quad D_2 = \bar{Q}_1 \oplus Q_2$$

$$y = \bar{x} \oplus Q_2 + x \oplus Q_1$$

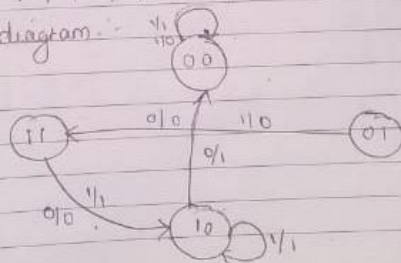
$$Q_1^+ = D_1$$

$$Q_2^+ = D_2$$

State table

| P.S | | x | N.S | | y |
|-------|-------|---|---------|---------|---|
| Q_1 | Q_2 | | Q_1^+ | Q_2^+ | |
| 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 |

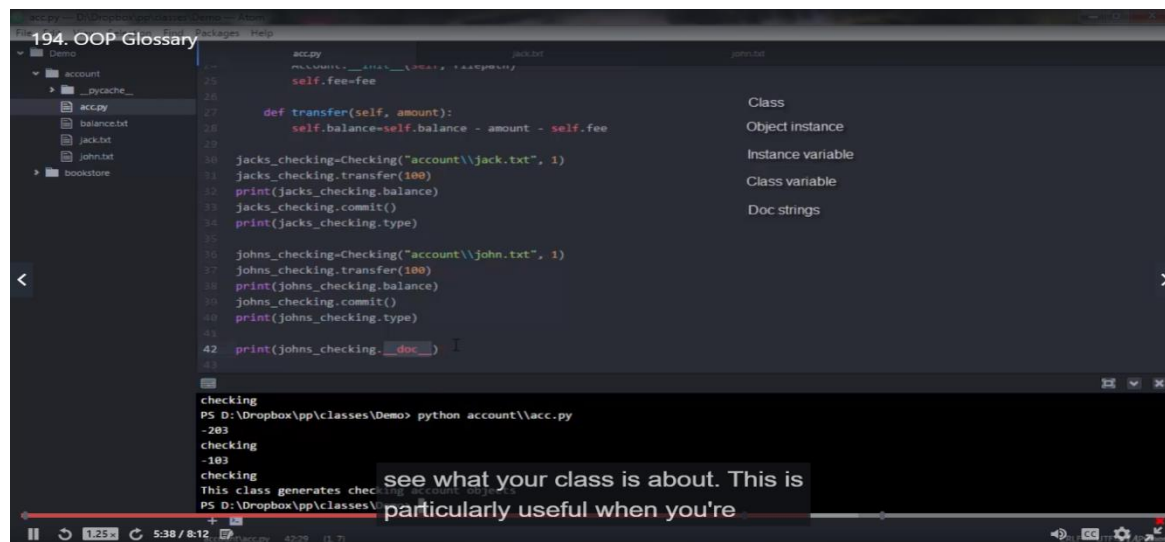
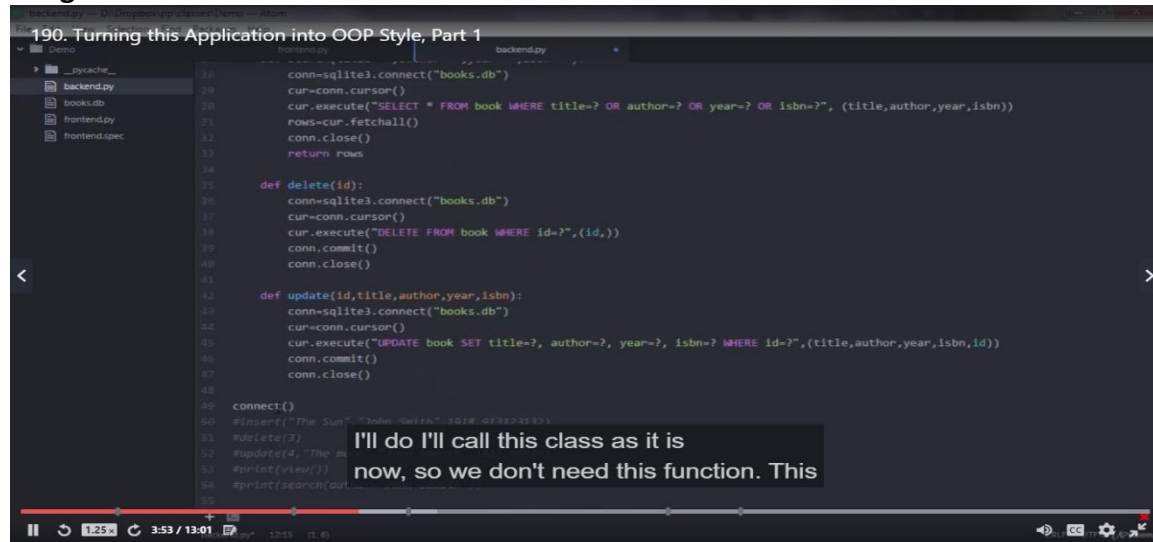
State diagram



| | | | |
|--------------------|------------------------------|--------------|---------------------|
| Date: | 29-05-2020 | Name: | Sushmitha r naik |
| Course: | Udemy-python | USN: | 4AL17EC090 |
| Topic: | 1.Object oriented programing | Sem and sec: | 6 th & B |
| Git hub repository | Sushmitha_naik | | |

AFTERNOON SESSION DETAILS

Image of session



Report – Report can be typed or hand written for up to two pages.

- The approach to solve a programming problem is by creating objects. This is known as Object-Oriented Programming (OOP).
- An object has two characteristics:

attributes

behavior

- After the introduction, we learnt on converting the frontend and backend designs and approaches involved for turning an application in OOP style.
- After that, we learnt about the different terminologies (glossary) involved in OOP python.

Inheritance:

Inheritance is a way of creating new class for using details of existing class without modifying it.

Class:

A user-defined prototype for an object that defines a set of attributes that characterize any object of the class.

Instance variable:

A variable that is defined inside a method and belongs only to the current instance of a class.

Object Instance:

An individual object of a certain class. An object obj that belongs to a class Circle, for example, is an instance of the class Circle.

Method:

A special kind of function that is defined in a class definition.

Instantiation:

The creation of an instance of a class.

Data member:

A class variable or instance variable that holds data associated with a class and its objects.

Constructor:

A constructor is a special kind of method that Python calls when it instantiates an object using the definitions found in your class.

Objects:

In python, functions too are objects. So, they have attributes like other objects.

