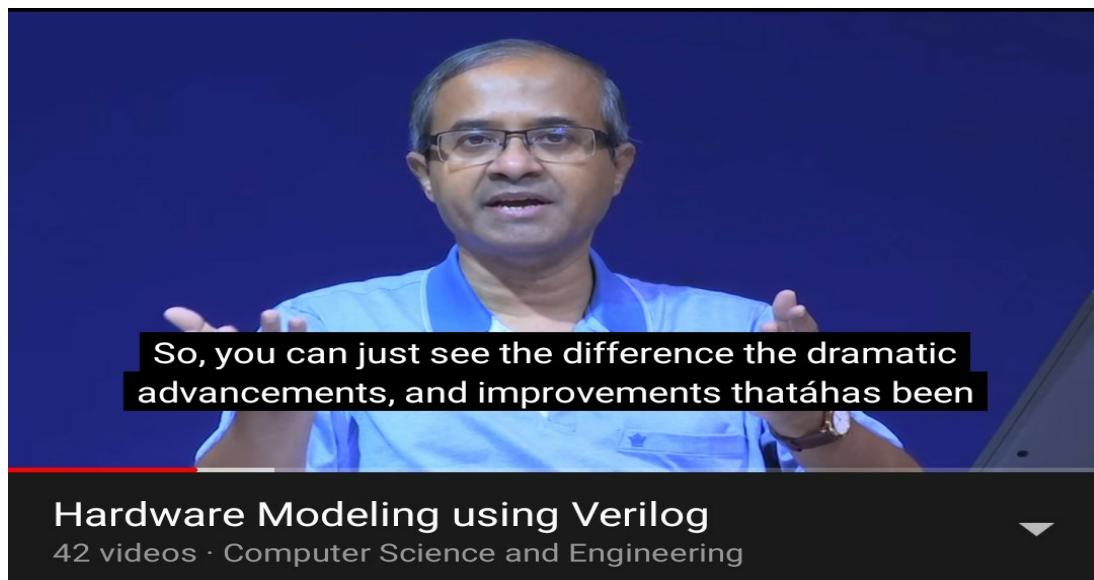


DAILY ASSESSMENT FORMAT

Date:	04-06-2020	Name:	K Muthu
Course:	DIGITAL DESIGN USING HDL	USN:	4a17ec038
Topic:	<ul style="list-style-type: none"> • Hardware modelling using verilog • FPGA and ASIC Interview questions • Task 	Semester & Section:	6 & 'A'
Github Repository:	K.Muthu-courses		

FORENOON SESSION DETAILS

Image of session



Introduction

41K views · 2 years ago



140



3



Share



Download



Save



Hardware Modeling Using Ve...

1.6K subscribers

SUBSCRIBE

Report –

Hardware modelling using verilog :

The outcome of the session are,

- ✓ Learnt about the verilog hardware description language.
- ✓ Understand the difference between behavioural and structural modelling styles.
- ✓ Learnt to write test benches and analyse simulation results.
- ✓ Learnt to model combinational logic circuits.
- ✓ Distinguish between good and bad coding practices.
- ✓ Case studies with some complex designs.

FPGA and ASIC Interview questions :

[Cmos FAQ](#)

[Misc FAQ](#)

[Home](#)

1) Write a verilog code to swap contents of two registers with and without a temporary register?

With temp reg ;

```
always @ (posedge clock)
begin
temp=b;
b=a;
a=temp;
end
```

Without temp reg;

```
always @ (posedge clock)
begin
a <= b;
b <= a;
end
```

[Click to view more](#)

Task : Implement a simple T Flipflop and test the module using a compiler.

The screenshot displays the EDA Playground web interface. On the left, a sidebar contains navigation links: "Languages & Libraries" (with sub-sections for Testbench + Design, UVM / OVM, and Other Libraries), "Tools & Simulators", and "Examples" (listing VHDL, Verilog/SystemVerilog, UVM, and EasierUVM). The main workspace is divided into two panels: "testbench.sv" and "design.sv".

The "testbench.sv" panel contains the following Verilog code:

```
1 // Code your SV/Verilog Testbench
2 // or browse Examples
3 module stimulus;
4   reg t,clk;
5   wire out;
6   t_ff t0 (Q,T,CLK);
7
8   initial
9   begin
10    CLK=1'b0
11    forever #5 clk=~clk
12    end
13
14   initial
15   $monitor($time,'t=%b,q=%b',T,
16           Q)
17
18   initial
19   begin
20    #10 T=1'b0;
21    #25 T=1'b1;
22    #10 T=1'b0;
23    #30 T=1'b1;
24    #60 $finish;
25   end
26 endmodule
```

The "design.sv" panel contains the following Verilog code for a T flip-flop:

```
1 // Code your SV/Verilog Design
2 module t_ff(q,t,clk):
3   input t,clk;
4   output out;
5   initial q=0
6   always@(posedge clk)
7   begin
8     if (t==1)
9       q=~q
10    else
11      q=0
12    end
13  end
14 endmodule
15
```

Date: 04-06-2020

Name: K Muthu

Course: Python Bootcamp 2020 build 15
working applications and Games

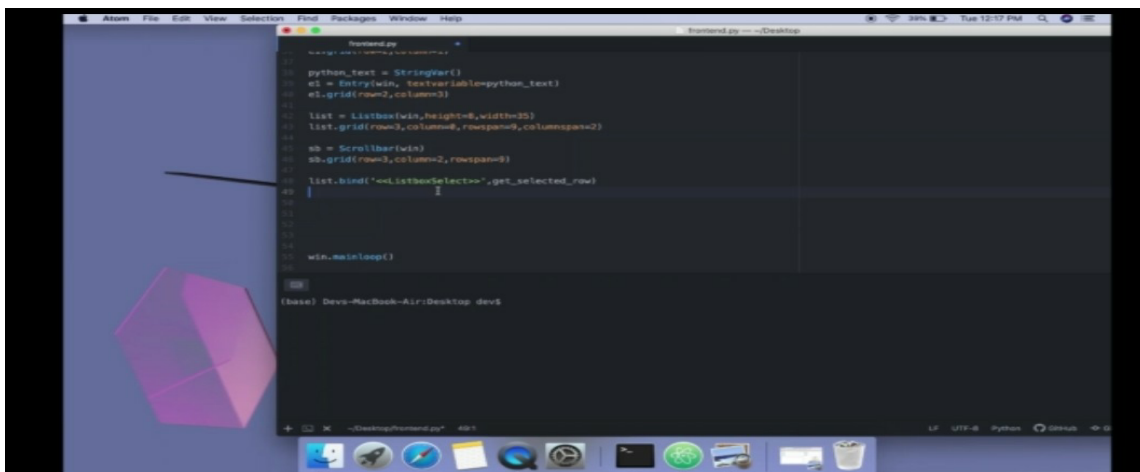
USN: 4a17ec038

Topic: Own database

Semester 6 & 'A'
& Section:

AFTERNOON SESSION DETAILS

Image of session



Lectures

More



Section 38 - Project-12 Your own database system



352 ☒ Introduction to this module

Video - 00:58 mins



353 ☒ Overview of project

Video - 04:06 mins



354 ☒ Labels and entries

Video - 04:20 mins



355 ☒ Entries, listbox and scrollbar



Report –

Own database :

- A database is an abstraction over an operating system's file system that makes it easier for developers to build applications that create, read, update and delete persistent data.
- are a concept with many implementations, including PostgreSQL, MySQL and SQLite.
- and MySQL are two of the most common open source databases for storing Python web applications' data.

Date :		Savings :	
Exercise :		Study :	
Diet :		Python :	

1 2-5-2020 300 {not done} {not studied}

2 15-6-2020 500 {not done} {not studied}

4 15-6-2020 500 {not done} {not studied}

7 20-6-2020 1500 {not done} {not studied}

10 3-5-2020 300 done studied {diet taken}

12 6-4-2020 2000 done studied {not taken}

13 8-4-2020 200 done studied {not taken}

ADD

SEARCH

DELETE

VIEW ALL

CLOSE