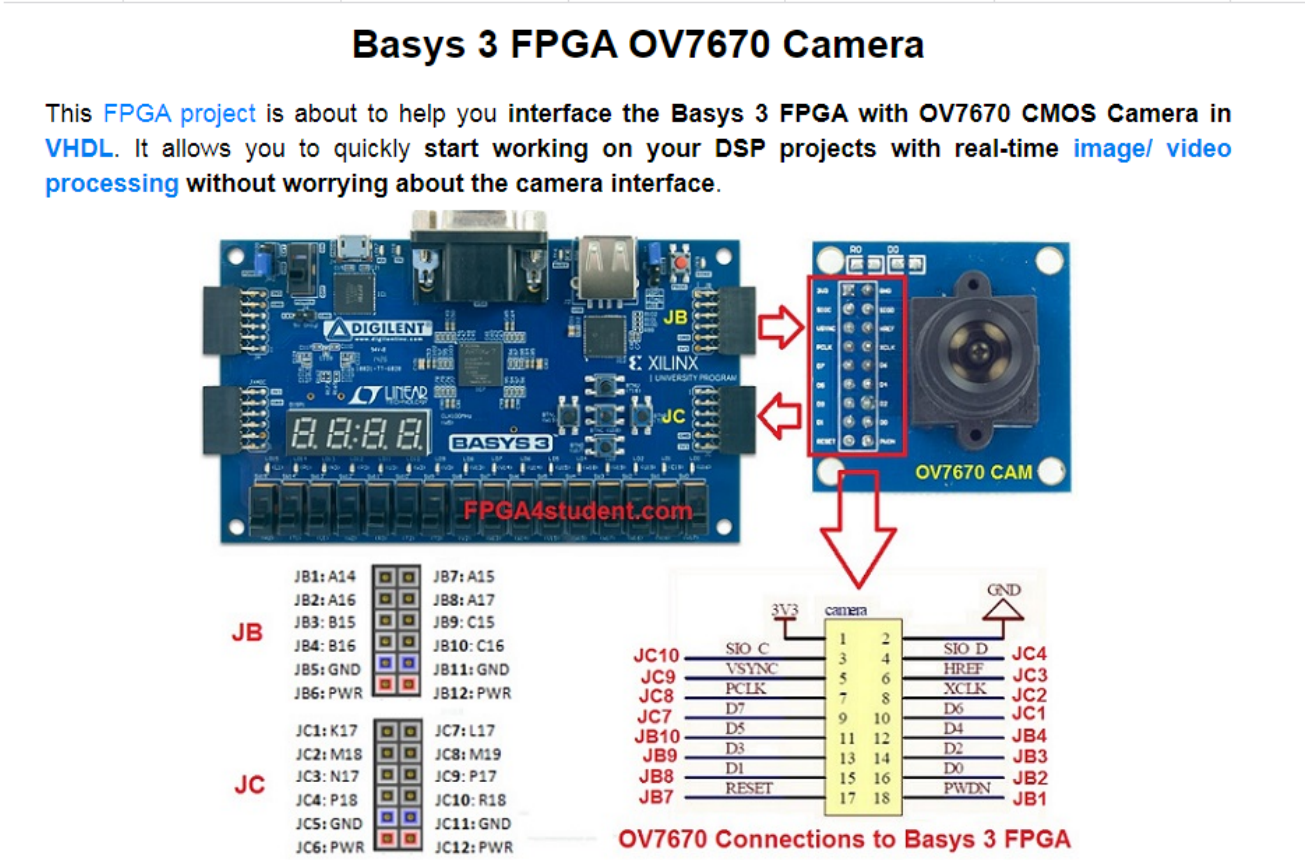
DAILY ASSESSMENT FORMAT

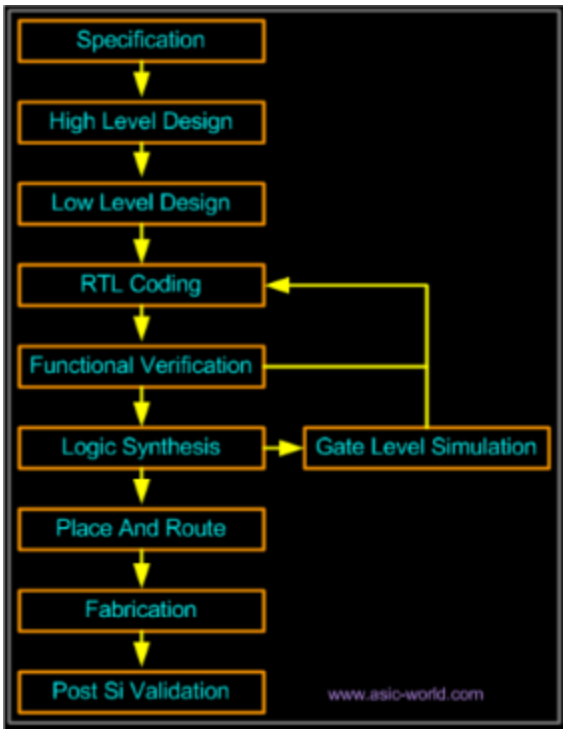
Date:	5/06/2020	Name:	Akshatha M Deshpande
Course:	DIGITAL DESIGN USING HDL	USN:	4AL17EC006
Topic:	Verilog Tutorials and practice programs,Building/ Demo projects using FPGA	Semester & Section:	6th Sem A sec
Github Repository:	AkshathaDeshpande		

FORENOON SESSION DETAILS
Image of session



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

Typical Design flow:



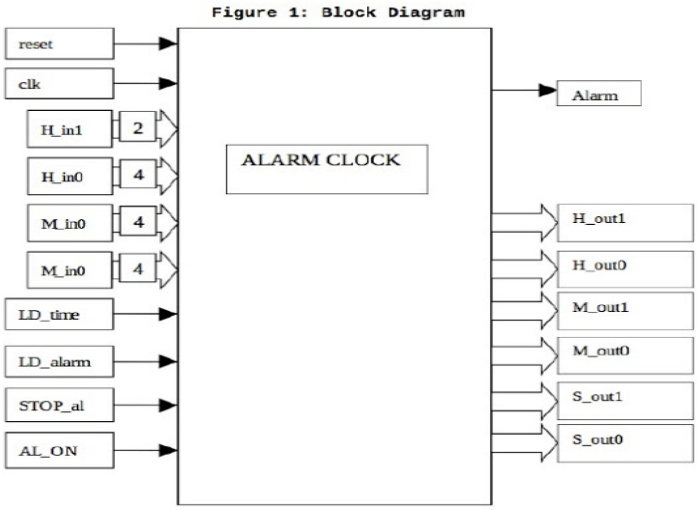
Simulation:

- Simulation is the process of verifying the functional characteristics of models at any level of abstraction.
- We use simulators to simulate the the Hardware models.
- To test if the RTL code meets the functional requirements of the specification, see if all the RTL blocks are functionally correct.
- To achieve this we need to write test bench, which generates clk, reset and required test vectors.
- A sample test bench for a counter is as shown below. Normally we spend 60–70% of time in verification of design.

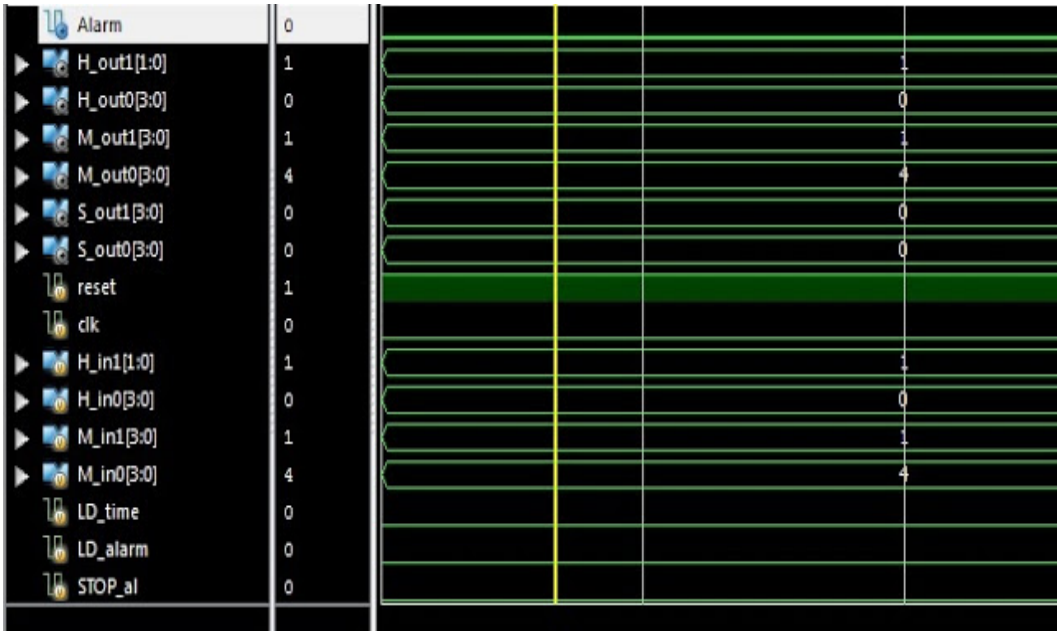
Place & Route:

Gate level netlist from the synthesis tool is taken and imported into place and route tool in Verilog netlist format. All the gates and flip–flops are places, Clock tree synthesis and reset is routed. After this each block is routed. Output of the P&R tool is GDS file, this files is used by foundry for fabricating the ASIC.

Alarm Clock:



Output:



- The simple alarm clock is shown in the following figure.
- The alarm clock outputs a real-time clock with a 24-hour format and also provides an alarm feature.
- Users also can set the clock time through switches.

Implement a verilog module to count number of 0's in a 16 bit number in compiler:

```
module num_zeros_for(
   input [15:0] A,
   output reg [4:0] ones
   );

integer i;

always@(A)
begin
   ones = 0;
   for(i=0;i<16;i=i+1)
      if(A[i] == 0'b1)
         ones = ones + 1;
end
endmodule
```

Output:

Input = "1011_0110_0010_0010"

Output = "01001"

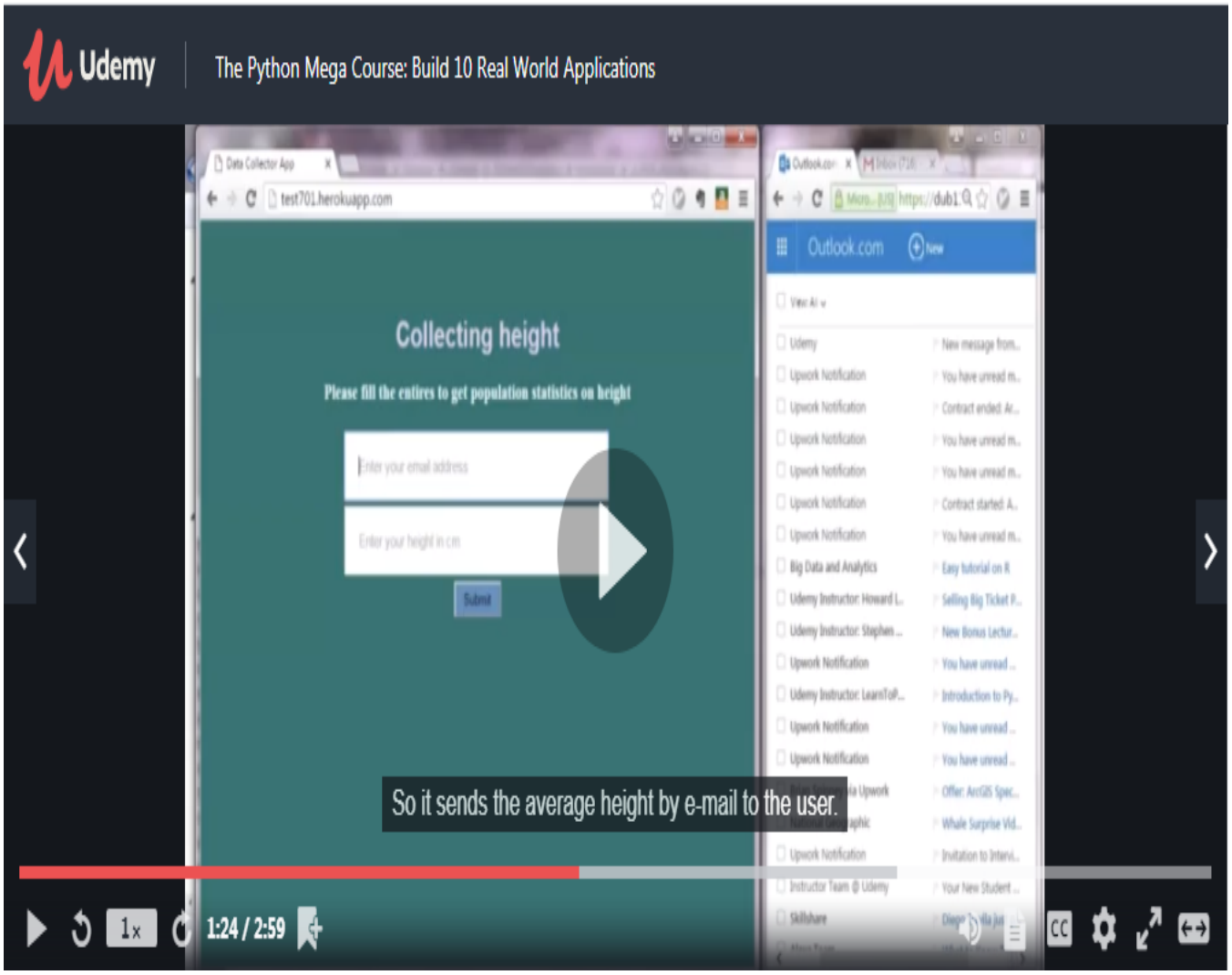
Input = "0001_0111_1010_0011"

Output = "01000" (8 in decimal)

Date:	5/06/2020	Name:	Akshatha M Deshpande
Course:	Python	USN:	4AL17EC006
Topic:	Application 10: Build a Data Collector Web App with PostGreSQL and Flask	Semester & Section:	6th Sem A sec

AFTERNOON SESSION DETAILS

Image of session



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

Application 10: Build a Data Collector Web App with PostGreSQL and Flask:

- Data Collector Web App - How The Output Will Look Like
- PostGreSQL Database Web App with Flask: Steps
- Frontend: HTML Part
- Frontend: CSS Part
- Backend: Getting User Input
- Backend: The PostGreSQL Database Model
- Backend: Storing User Data to the Database
- Backend: Emailing Database Values Back to the User
- Backend: Sending Statistics to Users
- Deploying the Web Application to a Live Server
- Bonus Lecture: Implementing Download and Upload in your Web App