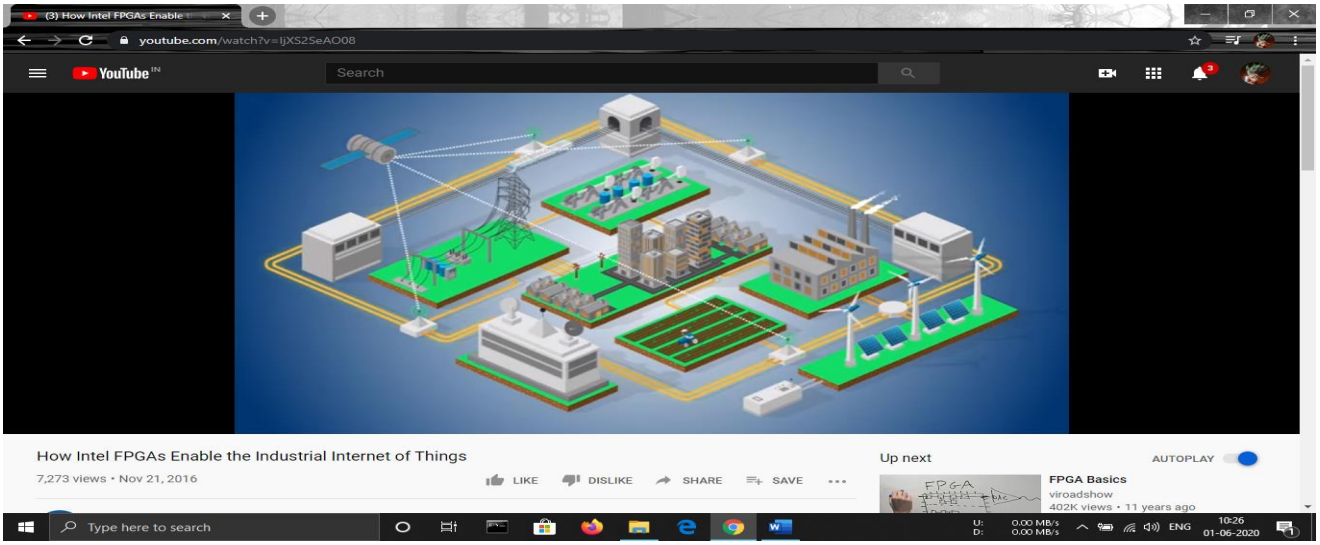
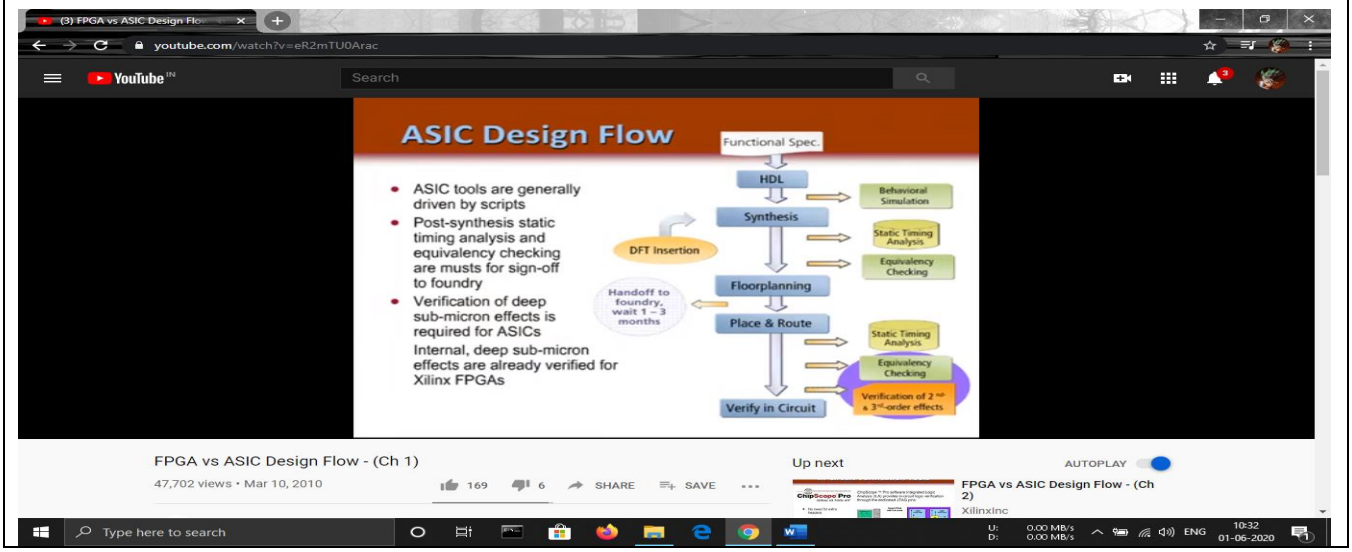
DAILY ASSESSMENT REPORT

Date:	**01/06/2020**	**Name:**	**Abhishek M Shastry K**
Subject:	**Digital Design Using HDL**	**USN:**	**4AL17EC002**
Topic:	**1] Industry Applications of FPGA** **2] FPGA Business Fundamentals** **3] FPGA vs ASIC Design Flow** **4] FPGA Basics – A Look Under the Hood**	**Semester & Section:**	**6th 'A'**
Github Repository:	**AbhishekShastry-Courses**		

FORENOON SESSION DETAILS

Image of session





Report

FPGA Business Fundamentals

- An **application-specific integrated circuit** (ASIC) is an integrated circuit (IC) chip customized for a particular use, rather than intended for general-purpose use.
- In computers, an **application-specific standard product** (ASSP) is a semiconductor device integrated circuit (IC) product that is dedicated to a specific application market and sold to more than one user.
- A **field-programmable gate array** (FPGA) is an integrated circuit designed to be configured by a customer or a designer after manufacturing.
- Advantages of ASIC/ASSP:
  - ✓ Low cost per unit (high volume mass production).
  - ✓ Low power consumption.
  - ✓ High performance/clock speed.
  - ✓ Small unit size.
- Disadvantages of ASIC/ASSP:
  - ✓ High non-recurring engineering cost (NRE).
  - ✓ Not flexible – cannot be upgraded once hardened.
  - ✓ Long time to market.

Core components of a FPGA

- LUT (Look-Up Table)
  - ✓ LUT is a logic lookup table, generally with 4 to 6 inputs and 1 to 2 outputs to specify any logical operation that fits within those bounds. There are however two other common uses for a LUT.
  - ✓ LUT as a shift register – shift registers are very useful for things like delaying the timing of an operation to align the outputs of one algorithm with another. Size varies based on FPGA.
  - ✓ LUT as a small memory – you can configure the LUT logic as a VERY small volatile random-access memory block. Size varies based on FPGA.

- FF (Flip-flop)
  - ✓ Flip-flops store the output of a combinational logic calculation. This is a critical element in FPGA design because you can only allow so much asynchronous logic and routing to occur before it is registered by a synchronous resource (the flip-flop), otherwise the FPGA won't make timing. It's the core of how an FPGA works.
  - ✓ Flip-flops can be used to register data every clock cycle, latch data, gate off data, or enable signals.
- Block Memory
  - ✓ This memory block is generally on the order of thousands of bits of memory, is configurable in width and depth, and multiple blocks of memory can be chained together to create larger memory elements.
  - ✓ They can generally be configured as either single-port or dual-port random access, or as a FIFO. There will generally be many block memory elements within an FPGA.
- I/O (Input/Output)
  - ✓ FPGAs will include I/O blocks that allow for various voltage standards (e.g. LVCMOS, LVDS) as well as timing delay elements to help align multiple signals with one another (e.g. for a parallel bus to an external RAM chip).

Future of FPGA with Intel

- **Accelerating Artificial Intelligence** - Machine learning pulls specifics from amount of data to predict and solve problems.
- **Accelerating Networks** - By driving fiber deep speeds into the network, FPGA's will increase throughput for higher bandwidth.
- **Accelerating Databases** - In databases, high performance FPGA's can extract maximum value from data analytics.
- **Accelerating the Data Center** - FPGA's accelerate the data center with light speed data transaction and storage processing to alleviate bottlenecks.

Task (DAY - 1)

Write a Verilog code to implement NAND gate in all different styles.

Gate Level modeling

```
module NAND_2_gate_level (output Y, input A, B);
   wire Yd;
   and (Yd, A, B);
   not (Y, Yd);
endmodule
```

Data flow modeling

```
module NAND_2_data_flow (output Y, input A, B);
   assign Y = ~ (A & B);
endmodule
```

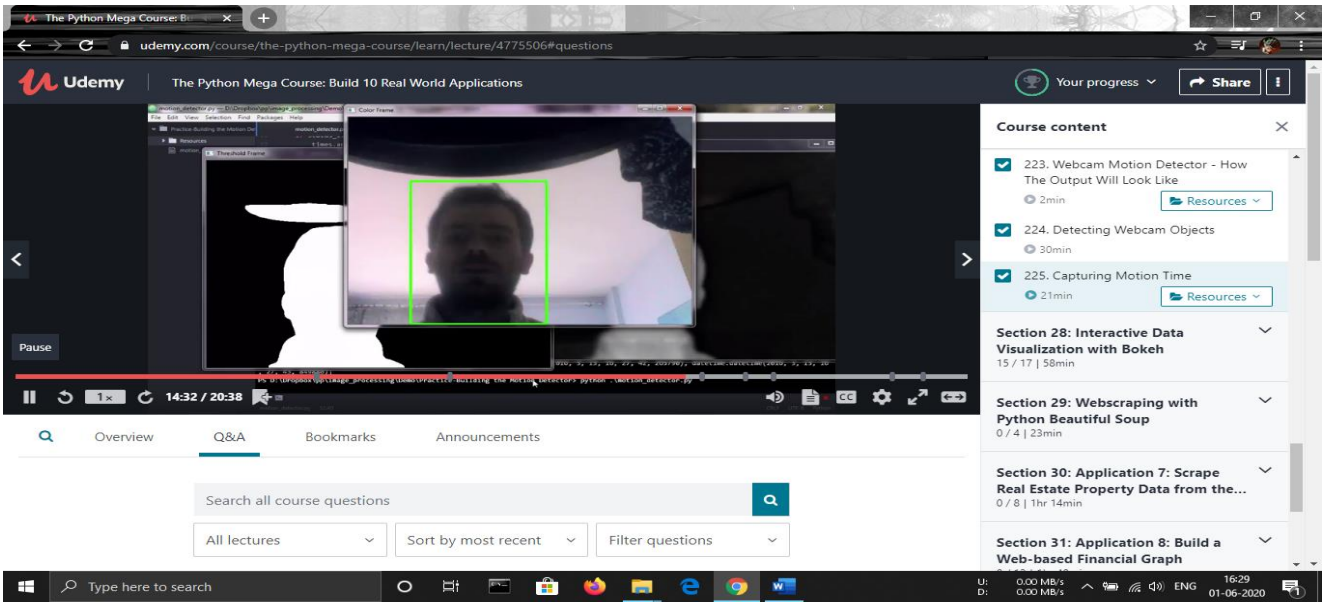
Behavioral Modeling

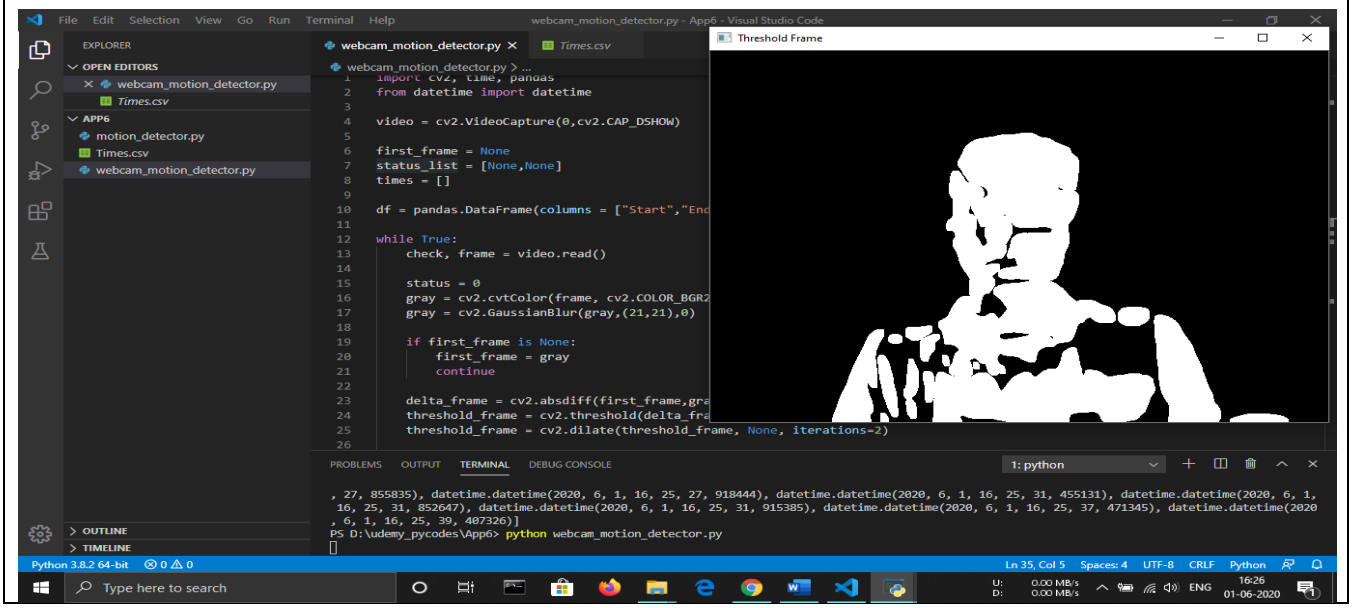
```
module NAND_2_behavioral (output reg Y, input A, B);
always @ (A or B) begin
   if (A == 1'b1 & B == 1'b1) begin
      Y = 1'b0;
   end
   else
      Y = 1'b1;
  end
endmodule
```

Date:	01/06/2020	Name:	Abhishek M Shastry K
Course:	The Python Mega Course: Build 10 Real World Applications	USN:	4AL17EC002
Topic:	1] Application 6: Build a Webcam Motion Detector	Semester & Section:	6th 'A'
Github Repository:	AbhishekShastry-Courses		

AFTERNOON SESSION DETAILS

Image of session





Report

Application 6: Build a Webcam Motion Detector

- Build a webcam motion detector using **OpenCV** (cv2 module) library for video capturing, **datetime** module for noting down the time and **pandas** library for storing data in a csv file.
- Some of the functions used under cv2 module:
  - ✓ Images also can contain different types of noise, especially because of the source (camera sensor). Image Smoothing techniques help in reducing the noise. OpenCV provides **cv2.GaussianBlur ()** function to apply Gaussian Smoothing on the input source image.
  - ✓ The function **cv2.absdiff ()** calculates the per-element absolute difference between two arrays or between an array and a scalar.
  - ✓ Thresholding is a technique in OpenCV, which is the assignment of pixel values in relation to the threshold value provided. In thresholding, each pixel value is compared with the threshold value. If the pixel value is smaller than the threshold, it is set to 0, otherwise, it is set to a maximum value (generally 255). For this technique **cv2.threshold ()** function is used.
  - ✓ The **cv2.dilate ()** function dilates an image by using a specific structuring element.
  - ✓ Contours are defined as the line joining all the points along the boundary of an image that are having the same intensity. Contours come handy in shape analysis, finding the size of the object of interest, and object detection. OpenCV has **cv2.findContour ()** function that helps in extracting the contours from the image. It works best on binary images, so we should first apply thresholding techniques, Sobel edges, etc.
  - ✓ The **cv2.boundingRect ()** function of OpenCV is used to draw an approximate rectangle around the binary image. This function is used mainly to highlight the region of interest after obtaining contours from an image.
- Pandas **pandas.DataFrame ()** function is a two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). Pandas DataFrame consists of three principal components, the data, rows, and columns.
- To write DataFrame to a csv file **pandas.to_csv ()** function is used.