DAILY ASSESSMENT

Date:	**1-6-2020**	**Name:**	**Kavyashree m**
Course:	**Digital design using HDL**	**USN:**	**4al15ec036**
Topic:	**Industry applications of FPGA,FPGA business fundamentals, FPGA Vs ASIC ,FPGA basics-A look under the hood**	**Semester & Section:**	**8th A**
Github Repository:	**kavya**		

FORENOON SESSION DETAILS

Image of session

Fig 1: Industry applications of FPGA

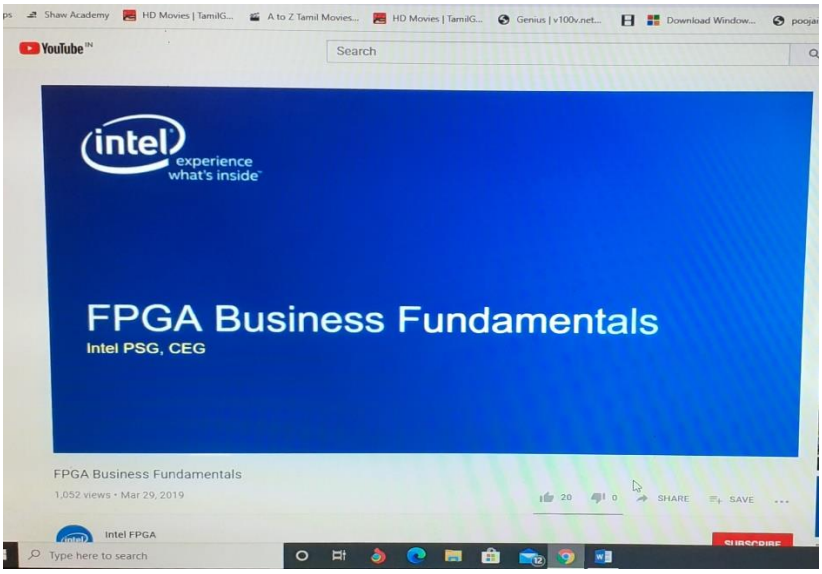


Fig 2 : FPGA business fundamentals

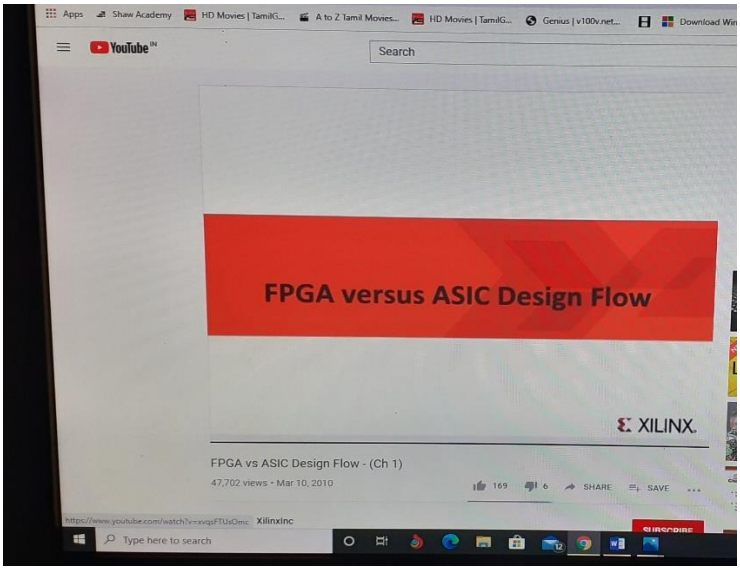


Fig 3 : FPGA Vs ASIC

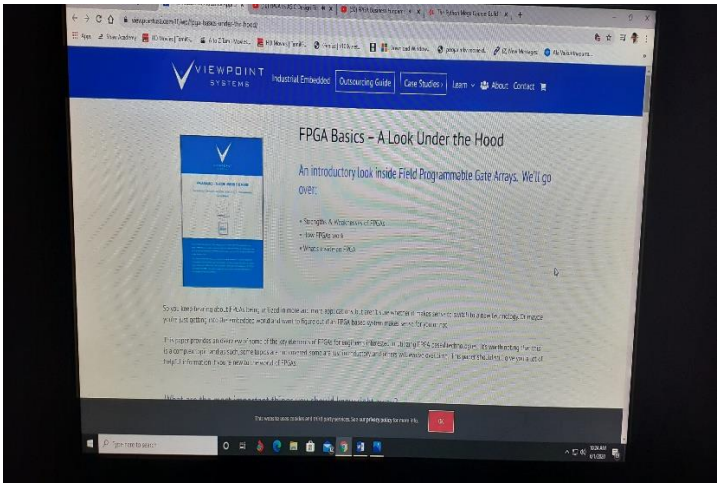


Fig 4: FPGA basics-A look under the hood

Industry applications of FPGA

Intel FPGAs solutions enable true flexibility and scalability to address IoT requirements with inherent hardware and software programmability. This powerful combination allows you to enable autonomous operation, tailor your solution to your customer's specific requirements and scale solutions to meet fragmented and evolving market requirements.

Smart City Infrastructure

At the heart of many Smart City applications are "intelligent vision" systems for environmental and weather monitoring, parking management, retail analytics and enhanced public safety. Intel FPGA SoCs enable high-performance and real-time video analytics for these applications.

Industry 4.0

Industry 4.0, or the fourth wave of the Industrial Revolution, leverages connectivity and autonomous operation to create robust industrial IoT applications. Our PLC solutions integrate M2M and secure enterprise application interoperability over cloud with OPC-UA, to deliver clear cost, power and TTM benefits for factory automation systems.

Smart Grid

We provide scalable, high-performance solutions to meet the requirements of today’s mission-critical Smart Grid functions, such as control loop, grid communications, network redundancy and security.

Healthcare systems

Healthcare systems can leverage IoT to enable fast and accurate aggregation of patient vital signs and other data from networked devices. These applications promise improved diagnostics and elimination of manual processes. FPGAs have the computing power to manage medical data aggregation, enabling improved patient care.

Data Center

The data center is central to the IoT as it processes data from millions of devices and sensors for highly optimized autonomous operation. Today, Intel® FPGAs are being used to improve throughput, response time and energy efficiency through the offloading of compute workloads from the server's central processing units onto FPGAs.

FPGA business fundamentals

The increase of logic in an FPGA has enabled larger and more complex algorithms to be programmed into the FPGA. The attachment of such an FPGA to a modern CPU over a high speed bus, like PCI express, has enabled the configurable logic to act more like a coprocessor rather than a peripheral. This has brought reconfigurable computing into the high-performance computing sphere.

Xilinx

Xilinx has developed two styles of partial reconfiguration of FPGA devices: module-based and difference-based. Module-based partial reconfiguration permits to reconfigure distinct modular parts of the design, while difference-based partial reconfiguration can be used when a small change is made to a design.

Intel

Intel supports partial reconfiguration of their FPGA devices on 28 nm devices such as Stratix V,and on the 20 nm Arria 10 devices. The Intel FPGA partial reconfiguration flow for Arria 10 is based on the hierarchical design methodology in the Quartus Prime Pro software where users create physical partitions of the FPGA that can be reconfigured at runtime while the remainder of the design continues to operate. The Quartus Prime Pro software also support hierarchical partial reconfiguration and simulation of partial reconfiguration.

FPGA Vs ASIC

Speed

ASIC rules out FPGA in terms of speed. As ASIC are designed for a specific application they can be optimized to maximum, hence we can have high speed in ASIC designs. ASIC can have hight speed clocks.

Cost

FPGAs are cost effective for small applications. But when it comes to complex and large volume designs ASIC products are cheaper.

Size/Area

FPGA are contains lots of LUTs, and routing channels which are connected via bit streams(program). As they are made for general purpose and because of re-usability. They are in-general larger designs than corresponding ASIC design. For example, LUT gives you both registered and non-register output, but if we require only non-registered output, then its a waste of having a extra circuitry. In this way ASIC will be smaller in size.

Power

FPGA designs consume more power than ASIC designs. As explained above the unwanted circuitry results wastage of power. FPGA wont allow us to have better power optimization. When it comes to ASIC designs we can optimize them to the fullest.

Time to Market

FPGA designs will till less time, as the design cycle is small when compared to that of ASIC designs. No need of layouts, masks or other back-end processes. Its very simple: Specifications -- HDL + simulations -- Synthesis -- Place and Route (along with static-analysis) -- Dump code onto FPGA and Verify. When it comes to ASIC we have to do floor planning and also advanced verification. The FPGA design flow eliminates the complex and time-consuming floor planning, place and route, timing analysis, and mask / re-spin stages of the project since the design logic is already synthesized to be placed onto an already verified, characterized FPGA device.

FPGA basics-A look under the hood

An introductory look inside Field Programmable Gate Arrays. We'll go over:

- Strengths & Weaknesses of FPGAs
- How FPGAs work
- What's inside an FPGA

What are the most important things you should know right away?

Get out of the software mindset – You're not writing software. Let me say that again because this is the single most important point if you're thinking about working with FPGAs.

You're designing a digital circuit. You're using code to tell the chip how to configure itself.

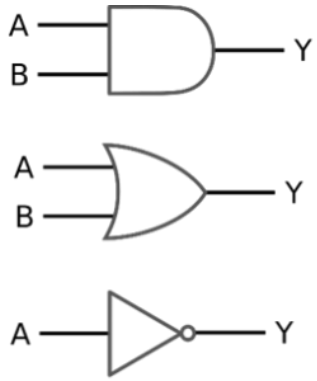
- Plan for lots of bugs – yes, plan for them. They are going to happen. Way more than you expected. If you're a newbie developer, you need to pull in someone that has experience with FPGA development to help with this estimate.
- Application-specific realities – you ought to concern yourself with realities revolving around cyber security and safety, as FPGAs are a different animal than what you're likely used to.

What is an FPGA?

An FPGA is a (mostly) digital, (re-)configurable ASIC. I say mostly because there are analog and mixed-signal aspects to modern FPGAs. For example, some have A/D

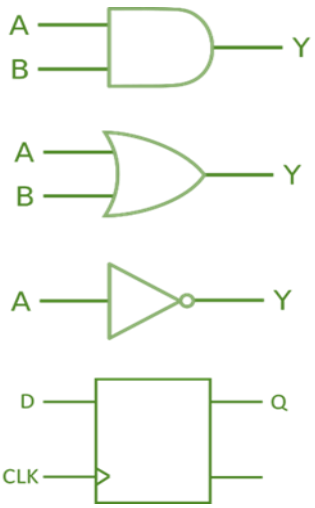
converters and PLLs. I put *re-* in parenthesis because there are actually one-time-programmable FPGAs, where once you configure them, that's it, never again. However, most FPGAs you'll come across are going to be re-configurable. So what do I mean by digitally configurable ASIC?

I mean that at the core of it, you're designing a digital logic circuit, as in AND, OR, NOT, flip-flops, etc. Of course that's not entirely accurate and there's much more to it than that, but that is the gist at its core.



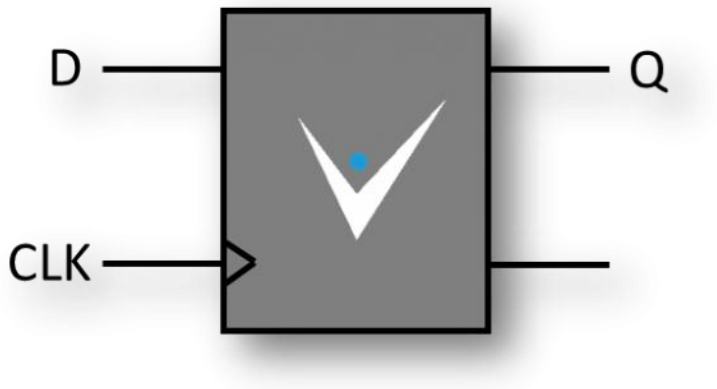
How Does an FPGA work?

You're designing a digital circuit more than anything else, basically at one layer of abstraction above the logic gate (AND, OR, NOT) level. At the most basic level, you need to think about how you're specifying the layout and equations at the level of LUTs (Look-Up Tables) and FFs (Flip-Flops).



Y = f(A,B)

A	B	Y
0	0	Y_1
0	1	Y_2
1	0	Y_3
1	1	Y_4



Otherwise you're circuit can get very large and slow very quickly. You've got a very detailed level of control at your fingertips, which is very powerful, but can be overwhelming, so start slow. You'll be determining the # of bits, and exact math / structure of each function.An FPGA is a synchronous device, meaning that logical operations are performed on a clock cycle-by-cycle basis. Flip-flops are the core element to enabling this structure.

What's Inside – Advanced components

Hard cores – These are functional blocks that (at least for the most part) have their own dedicated logical resources. In other words, they are already embedded into your FPGA silicon. You configure them with various parameters and tell the tools to enable them for you. This could include functions such as high-speed communications ,low-speed A/D converters for things like measuring slowly varying voltages, and microprocessor cores to handle some of the functions that FPGA logic is not as well suited for.

Soft cores – These are functional blocks that don't have their own dedicated logical resources. In other words, they are laid out with your core logic resources. You configure them with various parameters and tell the tools to build them for you. This

could include everything from DDR memory interfaces to FFT cores to FIR filters to microprocessors to CORDICs. The library of available soft cores can be impressive. On the plus side, you don't have to take as much time to develop these cores

AFTERNOON SESSION DETAILS

Date:	**1-6-2020**	**Name:**	**Kavyashree m**
Course:	**Python programming**	**USN:**	**4al15ec036**
Topic:	**Build a webcam motion detector**	**Semester & Section:**	**8th A**
Github Repository:	**kavya**		

Image of session

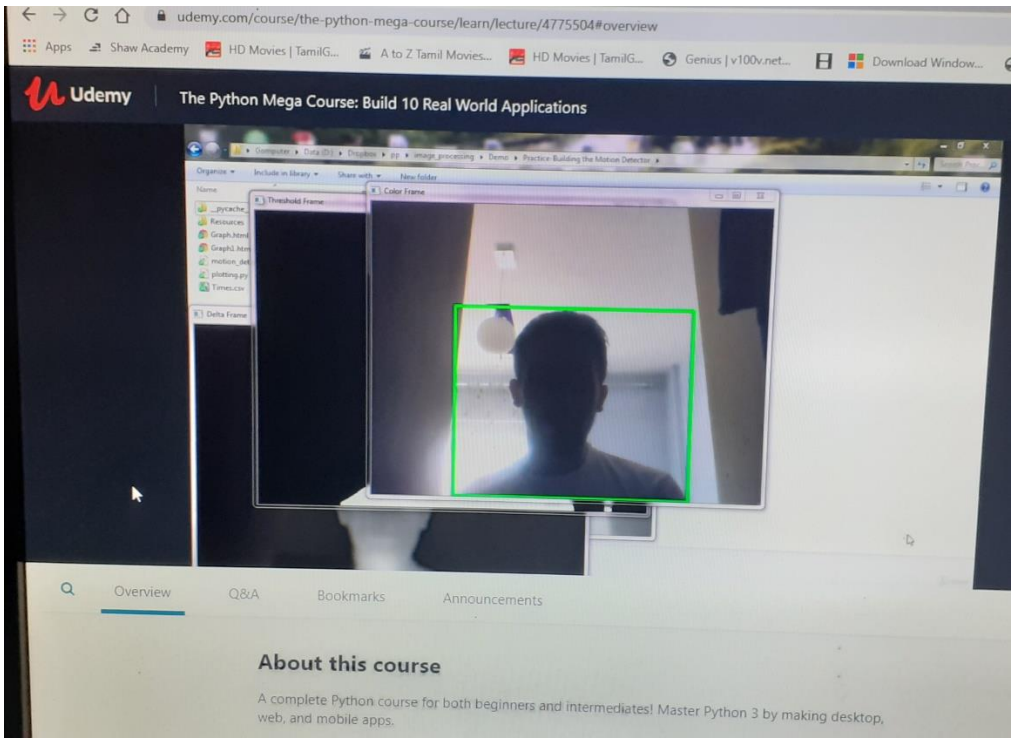


Fig 1: Build a webcam motion detector

WebCam Motion Detector in Python

This python program will allow you to detect motion and also store the time interval of the motion.

Requirement:

1. Python3

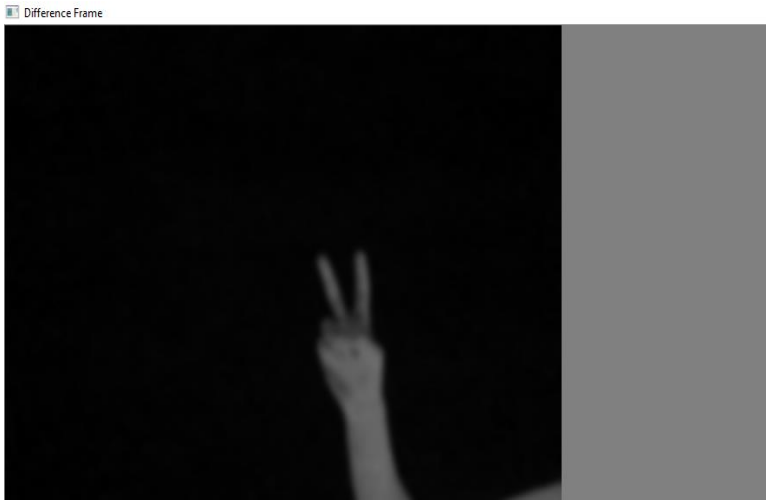
2. OpenCV(libraries)
3. Pandas(libraries)

Analysis of all windows

After running the code there 4 new window will appear on screen. Let's analyse it one by one:

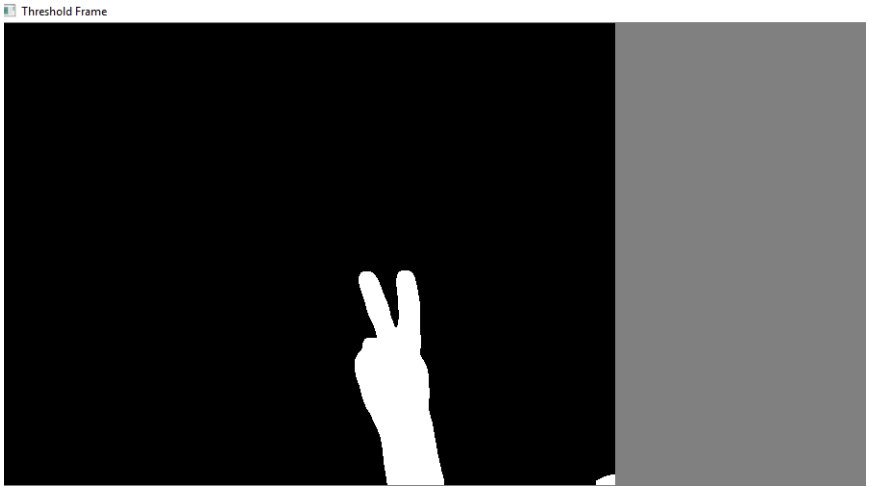
Gray Frame : In Gray frame the image is a bit blur and in grayscale we did so because, In gray pictures there is only one intensity value whereas in RGB(Red, Green and Blue) image thre are three intensity values. So it would be easy to calculate the intensity difference in grayscale.

Difference Frame : Difference frame shows the difference of intensities of first frame to the current frame.

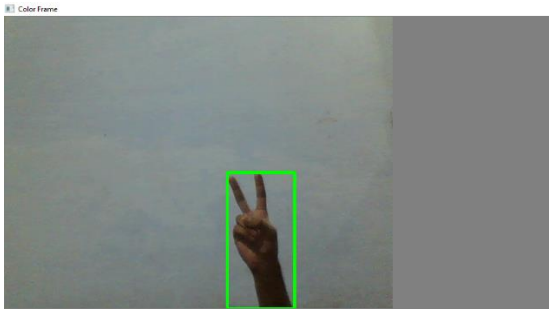


1. **Threshold Frame :** If the intensity difference for a particular pixel is more than 30(in my case) then that pixel will be white and if the difference is less than 30

that pixel will be black

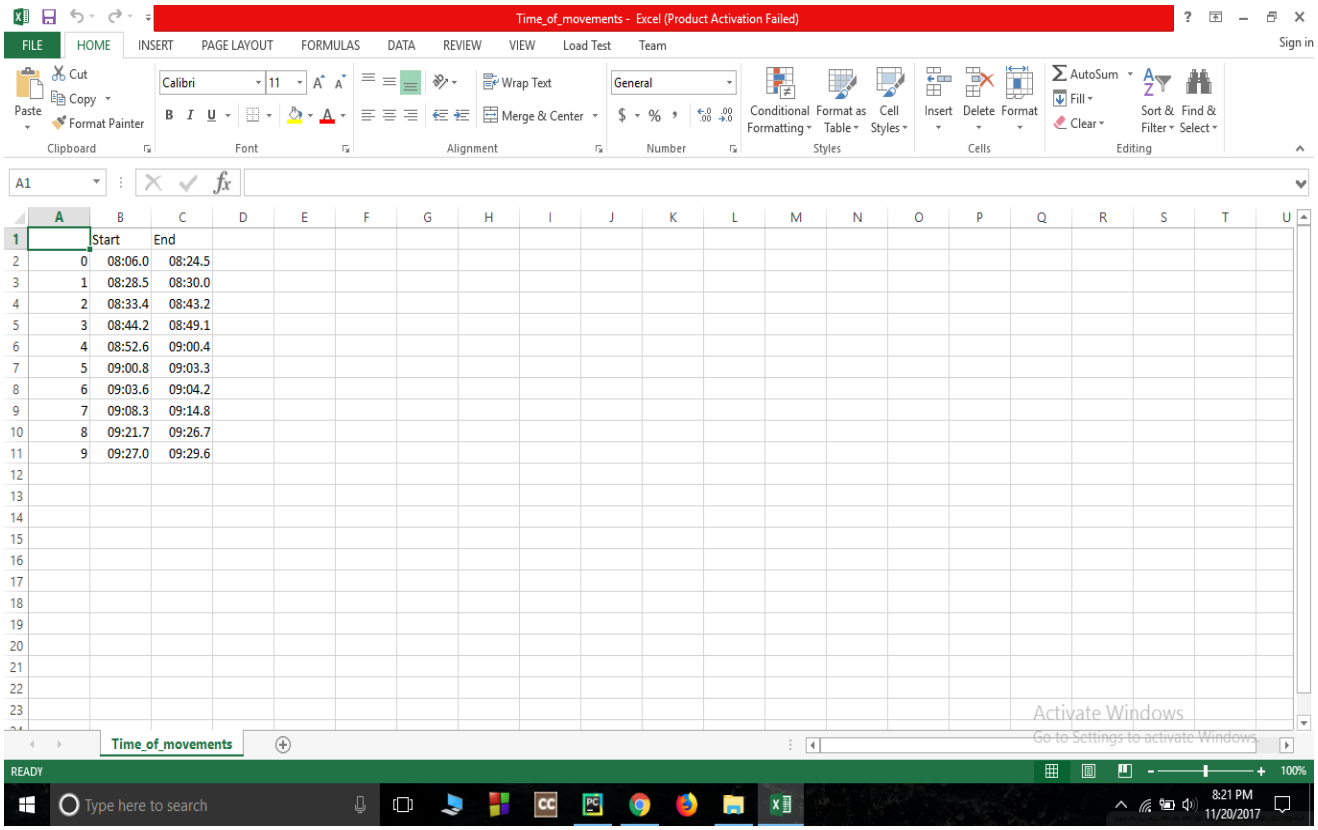


2. **Color Frame :** In this frame you can see the color images in color frame along with green contour around the moving objects



Time Record of movements

The Time of movements file will be stored in the folder where your code file is stored. This file will be in csv extension. In this file the start time of motion and the end time of motion will be recorded. As you can see in picture:


Time_of_movements - Excel (Product Activation Failed)
FILE
HOME
INSERT
PAGE LAYOUT
FORMULAS
DATA
REVIEW
VIEW
Load Test
Team
Sign in
Paste
Cut
Copy
Format Painter
Calibri
11
Wrap Text
Merge & Center
General
Conditional Formatting
Format as Table
Cell Styles
Insert
Delete
Format
AutoSum
Fill
Clear
Sort & Filter
Find & Select
Clipboard
Font
Alignment
Number
Styles
Cells
Editing
A1
Start
End
0 08:06.0 08:24.5
1 08:28.5 08:30.0
2 08:33.4 08:43.2
3 08:44.2 08:49.1
4 08:52.6 09:00.4
5 09:00.8 09:03.3
6 09:03.6 09:04.2
7 09:08.3 09:14.8
8 09:21.7 09:26.7
9 09:27.0 09:29.6
Activate Windows
Go to Settings to activate Windows.
Time_of_movements
READY
100%
Type here to search
8:21 PM
11/20/2017