

Daily Assessment Journal

Date: 30/01/2020

Name: Jyoti & Donus

Course: Digital design using HDL

UEN: 6A196037

Topic: Industrial applications of FPGA

- FPGA business fundamentals
- FPGA vs ASIC design flow

GitHub repository: jyoti-concepts

Forenoon session details

Report

The impact of new FPGA features in industrial application is analyzed in detail in three main areas, namely digital real-time simulation, advanced control techniques and electronic instrumentation, with focus on mechatronics, Robotics, & power system design.

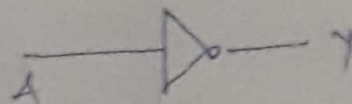
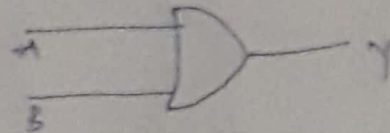
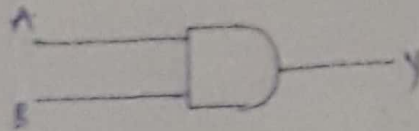
FPGA vs ASIC **FPGA**: A field programmable gate array is a semiconductor device containing programmable logic components called "logic blocks", & programmable interconnects. As ASIC are designed for a specific application they can be optimized to maximum, hence we can have high speed in ASIC designs.

How does an FPGA work? You're designing gate array (FPGA) digital ckt 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 & equations at the level of LUTs (Look-up tables) & FFs.

An FPGA is a digital, configurable ASIC. I say mostly because there are analog & mixed signal aspects to modern FPGAs. For ex: Some have ADC converters & PLLs. I put it in parenthesis because there are actually one time programmable FPGAs where once you configure it, it never changes 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 ckt, as in AND, OR, NOT, flipflops etc of course that's not entirely accurate & there's much more to it than that, but that is the gist at its core



Date: 01/06/2020

Course: python

Name: Jyoti. S. Dama
Vsn: 4AL17EC037

Topic: interactive data visualization
with Bokeh

Afternoon session details

image of session

Lecture 1: make a quick line graph with bokeh
from bokeh.plotting import figure, output_file, show
prepare some data

x = [1, 2, 3, 4, 5]

y = [6, 7, 8, 9, 7]

output_file("lines.html", title="line plot ex")

p = figure(title="simple line ex", xaxis_label='x', yaxis_label='y')

p.line(x, y, legend="Temp", line_width=2)

show(p)

Lecture 2: scatter plots

from bokeh.plotting import figure, output_file, show

output_file("line.html")

p = figure(plot_width=400, plot_height=400)

p.circle([1, 2, 3, 4, 5], [6, 7, 8, 9, 7], size=[1*2 for i in
[6, 7, 2, 4, 5]], color="blue", alpha=0.5)

Report

snippet producing the triangle based plot
making a basic bokeh line graph

```
# importing batch
from batchplotting import figure
from batch import output_file, show
```

```
# prepare some data
```

```
x = [8.7, 5]
```

```
y = [8.6, 9]
```

```
# prepare the output file
output_file("Line.html")
```

```
# create a fig object
```

```
f = figure()
```

```
f.triangle(x, y)
```

```
show(f)
```

```
# importing Batch
```

```
from batchplotting import figure
```

```
from batch import output_file, show
```

```
x = [8.7, 5]
```

```
y = [8.6, 9]
```

```
output_file("Line.html")
```

```
f = figure()
```

```
f.circle(x, y)
```

```
show(f)
```

```
# importing batch & pandas
```

```
from batchplotting import figure
```

```
from batch import output_file, show
```

```
import pandas
```

```
dq = pandas.read_csv("http://pythonhow.com/data/bachelors.csv")
```

```
x = dq["year"]
```

```
y = dq["engineering"]
```

```
output_file("Line from bachelors.html")
```

```
f = figure()
```



```

q.hline(x,y)
show(q)
import pandas
from bokeh.plotting import figure, output_file, show
df = pandas.read_excel("http://pythontab.com/data/verlages/
    herten.xlsx", sheet_name=0)
df["temperature"] = df["temperature"]/10
df["pressure"] = df["pressure"]/10
f = figure(plot_width=500, plot_height=400, tools='pan')
f.title.text = "temperature & air pressure"
f.title.text_color = "gray"
f.title.text_font_style = "bold"
f.xaxis.minor_tick_line_color = None
f.yaxis.minor_tick_line_color = None
f.xaxis.axis_label = "temperature(°C)"
f.yaxis.axis_label = "pressure(hPa)"
f.circle(df["temperature"], df["pressure"], size=0.5)
output_file("weather.html")
show(f)

```

Visual attributes

once you have built a basic plot, you can customize its visual attributes including changing the title color and font, adding labels for xaxis & yaxis, changing the color of the axis ticks, etc. All these properties are illustrated in the diagram below:

& here is the code if you want to play around with it

```

from bokeh.plotting import figure, output_file, show
f = figure(plot_width=500, plot_height=400, tools='pan, reset')
f.title.text = "Earthquakes"
f.title.text_color = "orange"

```

```
p.title_font="times"  
p.title.text_font_style="italic"  
p.yaxis.minor_tick_line_color="yellow"  
p.xaxis.axis_label="times"  
p.yaxis.axis_label="value"  
p.circle([1,2,3,4,5],[5,6,5,3,3],size=[1,2,4,15,5]  
color="red",alpha=0.5)  
output_file("scatter_plotting.html")  
show(p)
```

For a complete list of visual attributes see the styling
visual attributes documentation page of Bokeh.