Daily Assessment format

Name: Jyothi C Donna
USN: 4AL17EC037

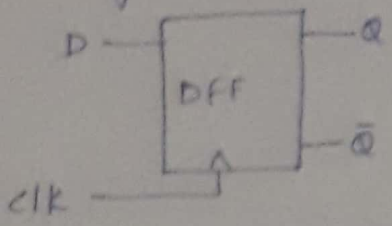
Date: 05/06/2020
Course: Digital design using HDL
Topic: Verilog tutorials & practice programs, building/Demo projects using FPGA
GitHub
Repository: Jyoth-course

forenoon session details

Image of session

Introduction

Verilog is a Hardware descriptional Language.




```
module d_ff_gate(q,q̄,d,clk);
input d,clk;
output q,qbar;
reg q,qbar;
always @(posedge clk)
begin
if(clk==1)
 q<=0;
qbar<=1;
else
 q<=d;
 qbar=!d;
end
endmodule
```

Report

Verilog tutorials

verilog is a Hardware description Language; a textual format for describing electronic ckts & systems. Applied to elect for verification through simulation, for timing analysis, for test analysis (testability analysis & fault sim) the verilog HDL is an IEEE standard-number 1364. the first version of the IEEE standard for verilog was published in 1995. A revised version was published in 2001; this is the version used by most verilog users. the IEEE verilog standard document is known as the language reference manual. authoritative definition of the verilog HDL. A further revision of the verilog standard was published in 2005.

FPGA

FPGA basics - A look under the hood an introductory look inside field programmable gate array. we'll go over; strengths & weaknesses of FPGAs how FPGAs work whats inside an FPGA so, you keep hearing about FPGAs being utilized in more & more applications, but aren't sure wheather it makes sense to switch to a new technology. or may be you're just getting into the embedde -d world & want to figure out if an FPGA based system makes sense for you. or not. this paper provides an overview of some of the key elements of FPGAs for engineers interested in topics are not covered. some are just introdu -ctory, & others will evolve over time. this paper should still give you a lot of helpful information if you're new to the world of FPGAs.

T-Flipflop

```
module tff(t,clk,q,qbar);
input t,clk;
output q,qbar;
reg q,qbar;
initial
begin
 q=0;
 q=1;
 end
always@(posedge(clk))
begin
if(t==0) q=q;
else
q=qbar;
qbar=~q;
end
endmodule
```

Date: 05/oct/2020
course: python
Topic: Build a data collector web app with postgresql & flask

Name: Jyoti. S. Donni
USN: 4AL17EC037

github
Repository: Jyoti-courses

Afternoon session details

Report:

Flask starting & configuration like most widely python libraries. the flask package is installable from the python package index(PPI). First create a directory to work in(something like flask_todo is a fine directory name) then install the flask package. you'll also want to install flask-sqlachemy so your flask application has a simple way to talk to a SQL database. A good way to get moving is to turn the codebase into an installable python distribution. At the project's root, create setup.py & a directory called todo to hold the source code. the setup.py should look something like this:

```
requires=[
        'flask',
        'flask-sqlalchemy',
        'psycopg2',
        ]

setup(
    name='flask_todo',
    version='0.0',
    description='A todo list built with flask',
    author='<your actual name here>',
    author_email='<your actual e-mail address here>',
    keywords='web flask',
    packages=find_packages(),
```

```
include_package_data=True,
install_requires=requires
)
```

That way, whenever you want to install or deploy your project, you'll have all the necessary packages in the requirements list. you'll also have everything you need to setup & install the package in site-packages. for more information on how to write an installable python distribution, check out the docs on setup.py within the todo directory containing your source code. create an app.py file & a blank __init__.py file. the __init__.py file allows you to import from todo as if it were an installed package. the app.py file will be the application's root. this is where all the flask application goodness will go, & you'll create an environment variable that points to that file. if you're using pipenv (like I am) you can locate your virtual environment with pipenv --venv & setup that environment variable in your environment's activate script.