On Desktop → Vivado 2017.4 → Double click the open

Quick Start → create project → Next →

New project → project name : project_today

project location : /home/student.

↳ Next

Project type : ⊙ RTL project ↳ Next

Add sources : Create file ↳ File type : verilog

File name : Andgate

file location : local to project

↳ OK

Name : Andgate.v → ~~create file~~ Next → Next

↳ product category : All

Family : Artix-7

Package : cpg236

Speed grade : -1

Temp grade : All remaining

part : xc7a35tcpg236-1

↳ Next

New project summary : It will give the summary of whatever is been selected

↳ finish

Define module : Module name : Andgate

I/O port definitions :

port name	direction
A	input
B	input
Y	output

↳ OK

Project Manager - project_today

project summary

Top module name: Andgate

Sources (window)

↳ Design sources

∴ Andgate (Andgate.v) ← double click

Andgate.v

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

→ ctrl S or Save

Sources

↳ Simulation sources → right click → Add sources →

Add sources → ⊙ Add or create simulation sources → Next

create file → File type : verilog

File name: Andgate_tb

File location: local to project

↳ ok → Finish

Module name: Andgate_tb → OK → Yes

Sources: ↳ simulation sources

↳ sim1

↳ ∴ Andgate (Andgate.v)

double click on → Andgate_tb (Andgate_tb.v)

Andgate_tb.v

```
module Andgate_tb ( );
reg A, B;
wire Y;
Andgate dut(A,B,Y);
```

```
initial
begin
A=0; B=0;
#5 A=0; B=1;
#5 A=1; B=0;
#5 A=1; B=1;
end
endmodule
```

dut: device under test

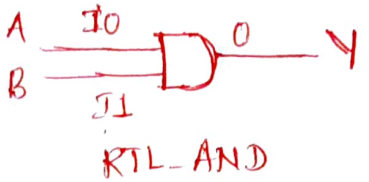
#5 : delay & time

→ Save

Flow Navigator
↳ RTL Analysis
↳ Open elaborated design → OK

Flow Navigator
↳ RTL Analysis
↳ Schematic

Schematic



This is the RTL schematic of AND gate

Flow Navigator
↳ Simulation
↳ Run simulation → Run Behavioral simulation

We will get the window where we can see and verify the waveform for 2 i/p AND gate

Flow Navigator
↳ Synthesis
↳ Run Synthesis

Synthesis completed
↳ ⊙ open synthesized design → OK

Flow Navigator
↳ Synthesis
↳ schematic

We will get the technology schematic of AND gate

⊙ single click on technology schematic

cell properties ↳ Truth table → We can see the Truth table of 2 i/p AND gate

On right most top → I/o planning

Select I/o ports

↳ double click on scalar ports

Scalar ports

A	IN	R2	LVCMOS33
B	IN	T1	LVCMOS33
Y	OUT	L1	LVCMOS33

R2 & T1 → Number of first two inputs on board

L1 → Number of first LED o/p.

↳ ctrl S (Save) → Ok

Save Constraints:

File type: XDC (xylinx Design constraint file)

File name: Andgate.xdc

File location: Local to project → Ok

Synthesized design:

Sources

↳ Constraints

↳ Constr_1(1)

↳ Andgate.xdc (target)

↑ double click to get xdc file

xdc file will show the i/p & o/p pins selected on board.

Flow Navigator

↳ Implementation

↳ Run implementation → Yes

Connect the BASYS 3 board to CPU
switch on the board

Implementation completed
↳ ⊙ Generate Bitstream → OK

Bitstream Generation successfully completed
↳ ⊙ Open Hardware Manager → OK

Hardware manager
↳ Open target → [illegible]
↳ Auto connect

Hardware Manager
↳ program device
↳ select the Andgate.bit file
↳ program

verify the o/p on board using selected input switches & o/p LED's.

To create one more file under the same project :

close the Hardware manager

Remove the .xdc files already created as:

Sources
↳ Constraints
↳ Constrs_1
↳ Andgate.xdc (target)

right click on Andgate.xdc (target)
↳ Remove file from project → Ok

ruby do for any other .xdc files.

close synthesized design
close elaborated design
close the main prog & test bench

Project Manager - project_today

Sources
↳ Design sources → right click → Add sources

Add sources:
↳ ⊙ Add or create design sources
↳ Next

Add so → create file → File type : verilog
File name: Orgate
File location: local to project
↳ Ok

Name : Orgate.V → Finish

Define module: module name : Orgate

A	I/P
B	I/P
Y	O/P

↳ Ok

Project Manager - project_today

↓

Sources ↳ Design sources

↳ ∴ Andgate (Andgate.v)

Orgate (Orgate.v)

right click on Orgate (Orgate.v) → set as top

→ Design sources

↳ ∴ Orgate (Orgate.v)

Andgate (Andgate.v)

ie Orgate is set as the top module which can be seen in the project summary as:

TOP module name: Orgate

double click on ∴ Orgate (Orgate.v)

Orgate.v

assign Y = A|B; → save

Sources ↳ simulation sources → right click → Add sources →

Add source → (1) Add or create simulation sources → Next

create file → File type : verilog

file name: Orgate_tb

file location: local to project

↳ ok → Finish

Module name: Orgate_tb → ok → Yes

Sources ↳ Simulation sources ↳ Sim 1

↳ ∴ Andgate_tb

Orgate_tb

right click on Orgate_tb → set as top

↳ ∴ Orgate_tb (Orgate_tb.v) → double click

write test bench file

Program to verify the truth table of all basic and universal gates

port name	Direction	Bus	MSB	LSB
A, B	i/p			
Y	o/p	~	7	0

→ ok

```
Module Allgates (input A,B, output [7:0] Y);
    assign Y[0] = A & B;        AND gate
    assign Y[1] = A | B;        OR gate
    assign Y[2] = ~A;           NOT gate
    assign Y[3] = A ^ B;        XOR gate
    assign Y[4] = ~(A & B);     NAND
    assign Y[5] = ~(A | B);     NOR
    assign Y[6] = ~(A ^ B);     XNOR
    assign Y[7] = A;            Buffer
endmodule
```

Allgates_tb.v

```
Module Allgates_tb( );
  reg A, B;
  Wire [7:0] Y;
  Allgates dut (A,B,Y);
  initial
  begin
  A=0; B=0;
#5 A=0; B=1;
#5 A=1; B=0;
#5 A=1; B=1;
  end  endmodule
```

Scalar ports

A	i/p	R2	LVCMOS33
B	i/p	T1	
Y[0]	OUT	L1	LVCMOS33
Y[1]	OUT	P1	
Y[2]	OUT	N3	
Y[3]	OUT	P3	
Y[4]	OUT	U3	
Y[5]	OUT	W3	
Y[6]	OUT	V3	
Y[7]	OUT	V13	

Note:

Flow Navigator
↳ Synthesis
↳ Report Utilization
↳