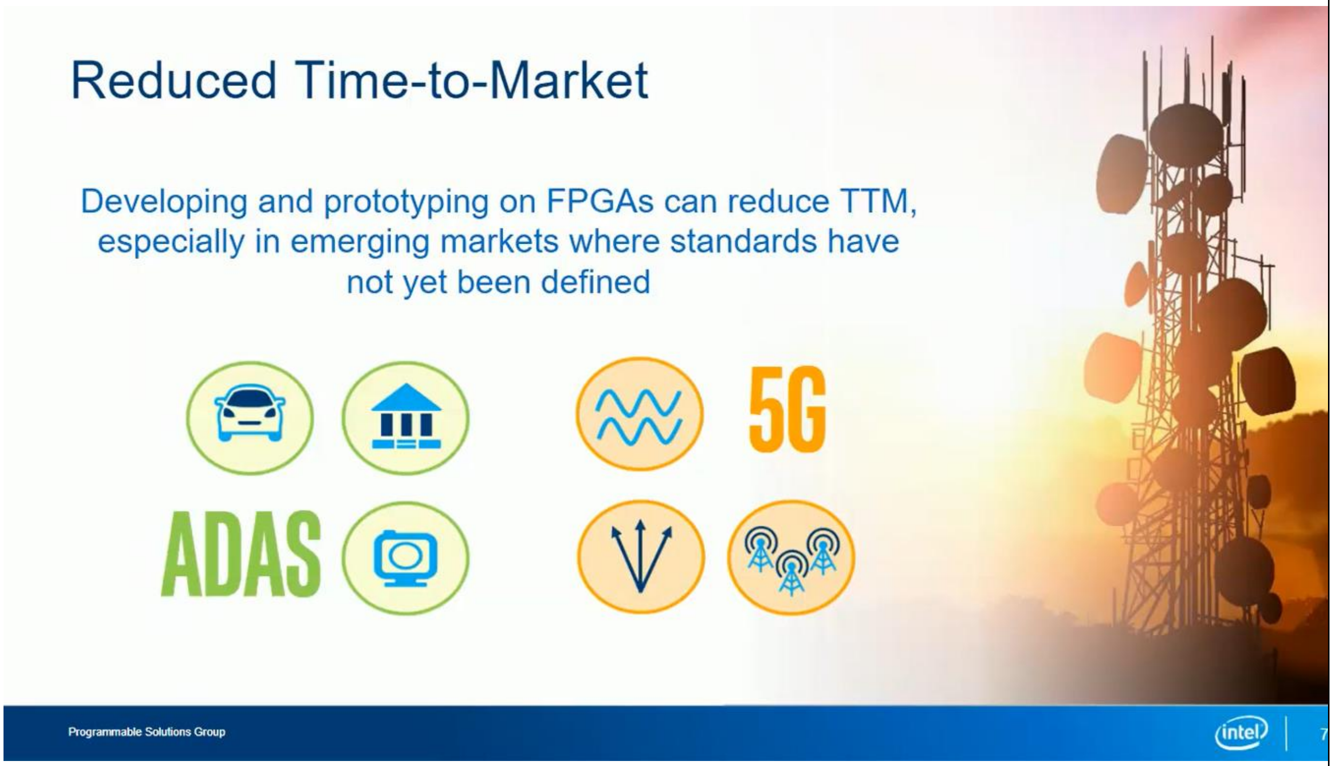
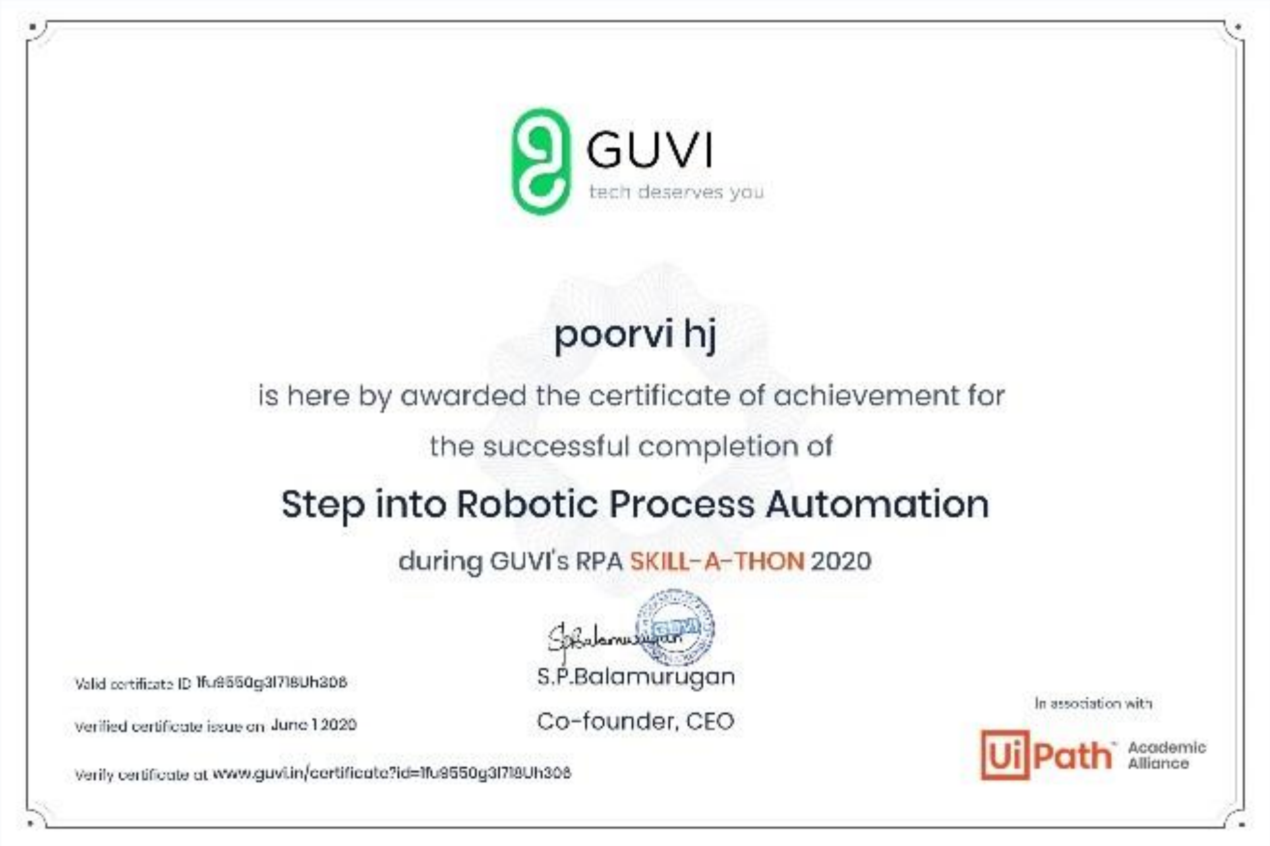
DAILY ASSESSMENT FORMAT

Date:	**01/06/2020**	**Name:**	**Poorvi hj**
Course:	**Digital design using HDL**	**USN:**	**4al17ec086**
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 , Bsec**
Github Repository:	**Poorvi-2000**		

FORENOON SESSION DETAILS
Image of session  Reduced Time-to-Market Developing and prototyping on FPGAs can reduce TTM, especially in emerging markets where standards have not yet been defined



GUVI

tech deserves you

poorvi hj

is here by awarded the certificate of achievement for

the successful completion of

Step into Robotic Process Automation

during GUVI's RPA SKILL-A-THON 2020

GUVI

S.P.Balamurugan

Co-founder, CEO

Valid certificate ID 1fu9550g3l718Uh308

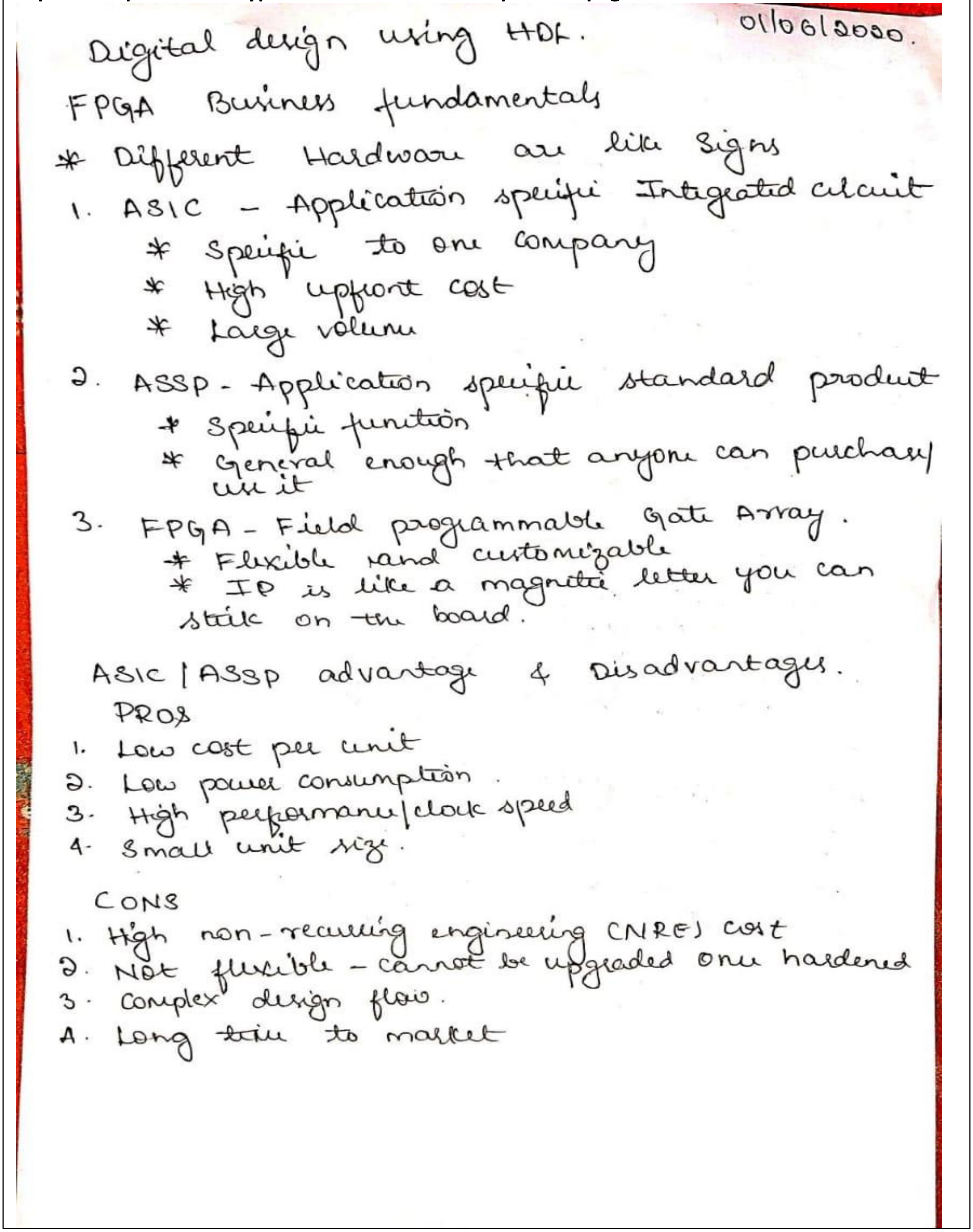
Verified certificate issue on June 1 2020

Verify certificate at www.guvi.in/certificate?id=1fu9550g3l718Uh308

In association with

UiPath Academic Alliance

Report – Report can be typed or hand written for up to two pages.m



Digital design using HDL. 01/06/2020.

FPGA Business fundamentals

* Different Hardware are like signs

1. ASIC – Application specific Integrated circuit
   * Specific to one company
   * High upfront cost
   * Large volume
2. ASSP - Application specific standard product
   * Specific function
   * General enough that anyone can purchase/use it
3. FPGA - Field programmable Gate Array.
   * Flexible and customizable
   * IP is like a magnetic letter you can stick on the board.

ASIC/ASSP advantage & Disadvantages.

PROS

1. Low cost per unit
2. Low power consumption.
3. High performance/clock speed
4. Small unit size.

CONS

1. High non-recurring engineering (NRE) cost
2. Not flexible - cannot be upgraded once hardened
3. Complex design flow.
4. Long time to market

Why FPGA

1. Reprogrammable & flexible
2. Product Longevity
3. Reduced time to market
4. market-size optimized.

FPGA v/s ASIC design flow.

Design flow.

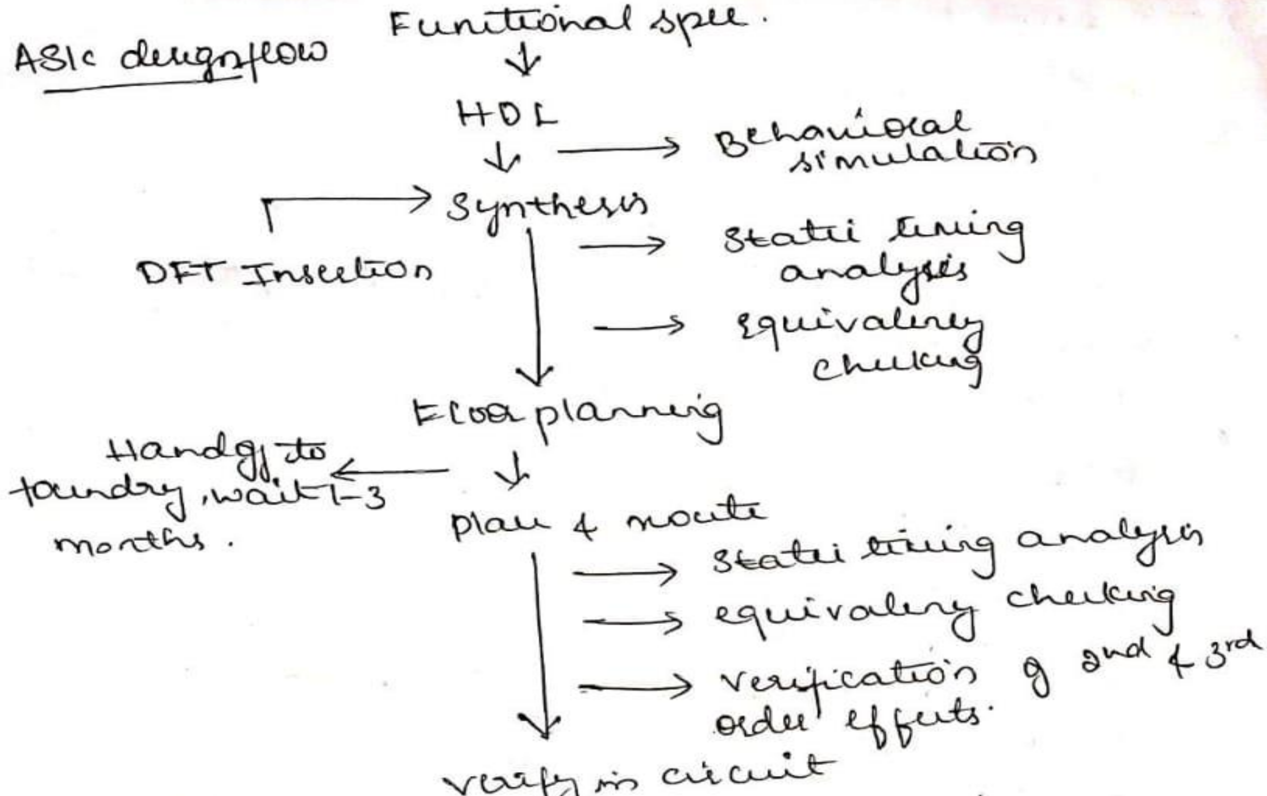
1. Asic & FPGA design & implementation
methodologies differ somewhat
2. Xilinix FPGA provide for reduced design
time and latter bug fixes.
   - No design for test logic is required
   - Deep sub-micron verification is done
   - No waiting for prototypes

3. coding style
   - For high performance designs, FPGAs may
   require some pipelining
   - When retargeting code from an ASIC to
   an FPGA, the code usually requires optimi-
   -zation

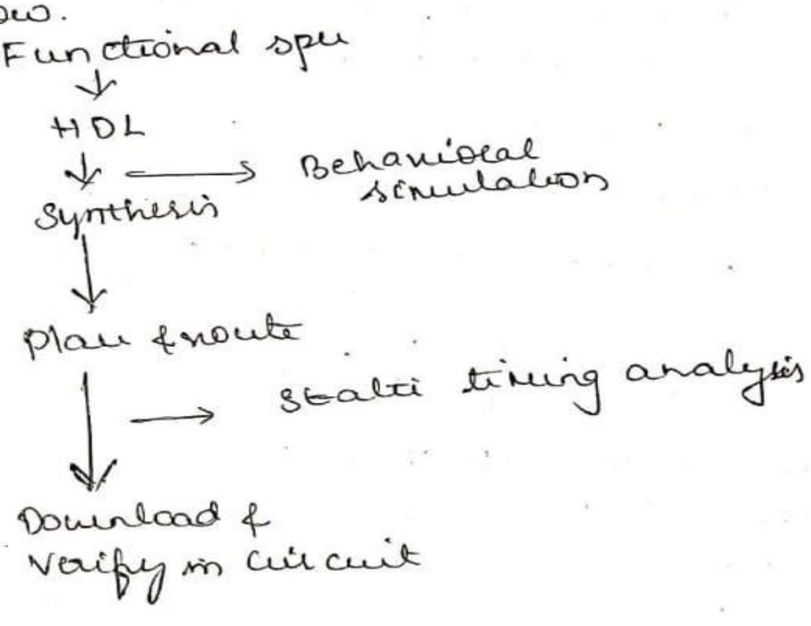
ASIC design flow.

1. ASIC tools are generally driven by scripts
2. post synthesis static timing analysis and
equivalency checking are musts for sign-off
to foundry.

ASIC designflow



FPGA design flow.



ASIC implementation

1. Create HDL.
2. Synthesis
   - Primarily driven by scripts
   - Synopsys design compile
   - Design for test logic insertion [BIST, Scan and JTAG]
3. Place & route
   - Foundry tools, cadence, AVANT
4.

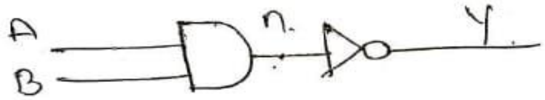
FPGA

1. Create HDL.
2. Synthesis
   - Synopsys, mentor, XST
   - Pushbutton flow with scripting capabilities
3. Place & route
   - completed by the user
   - Xilinx implementation tool - ISE software.
   - Pushbutton flow, scripting capabilities.

FPGA Basics:

FPGA is a digital re-configurable ASIC.

Implementation of NAND gate.



```
modulem(A,B,Y);
input A,B;
output Y;
wire n.
and(n,A,B);
not (n,Y);
endmodule
```

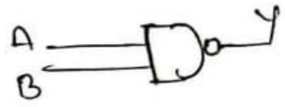
Data flow modelling.

Y = n(A&B).

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

Behavioral modeling

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



A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0