

Text editor

openxxr

micro

VScode

Sublime text

Tool

Niradlo

Quartus adv

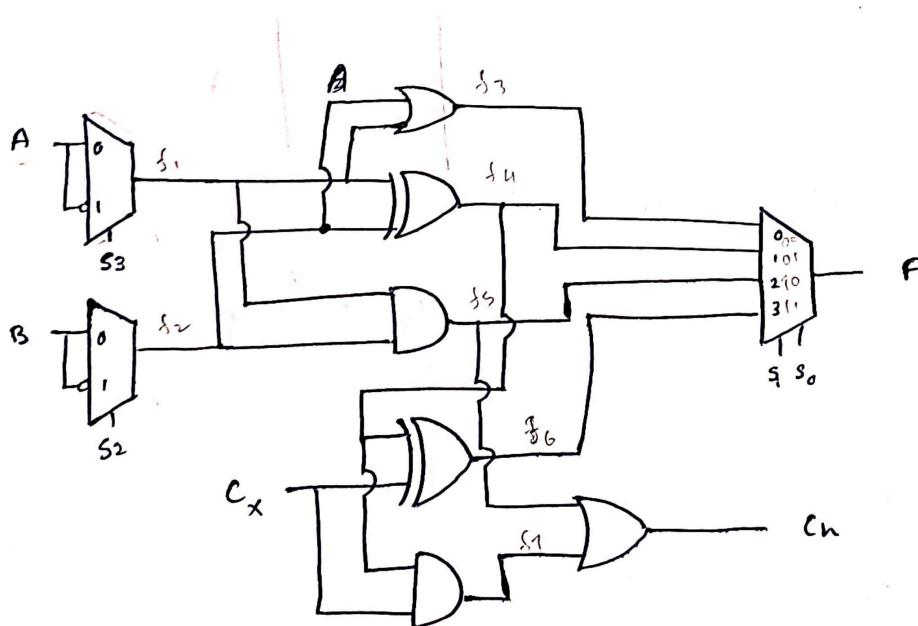
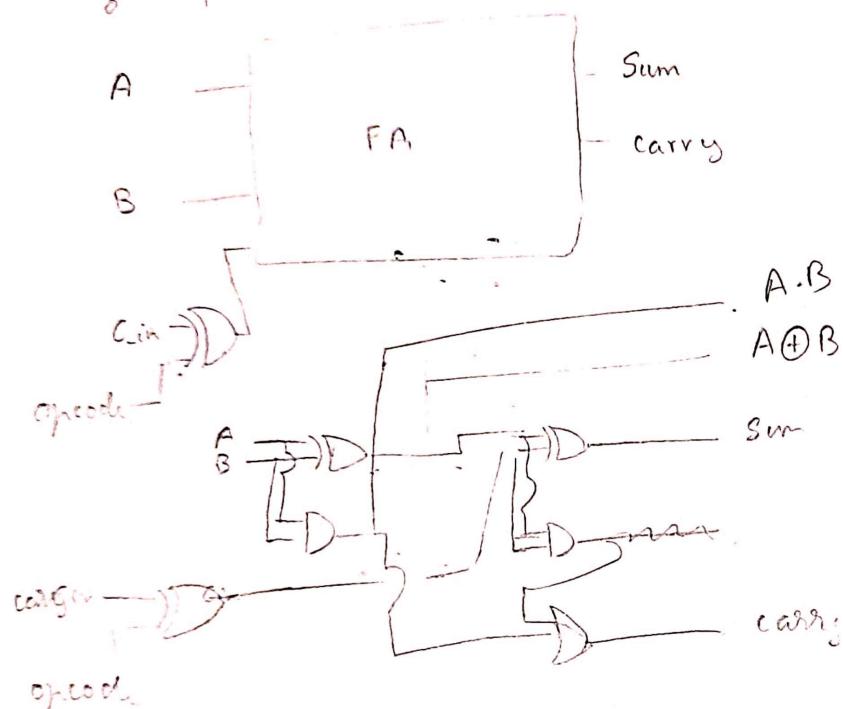
Fosgs

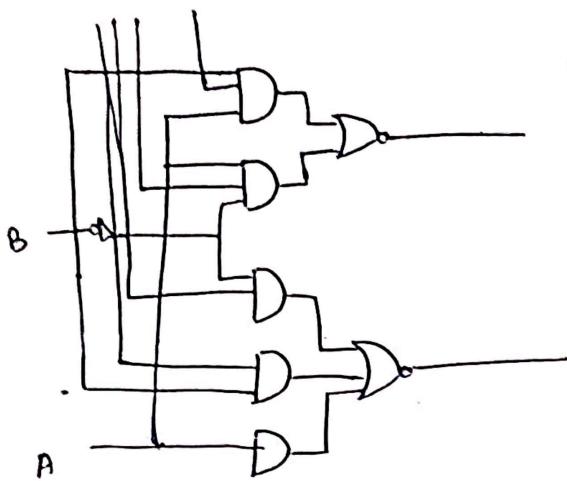
IEEE 1800:

VHDL 93

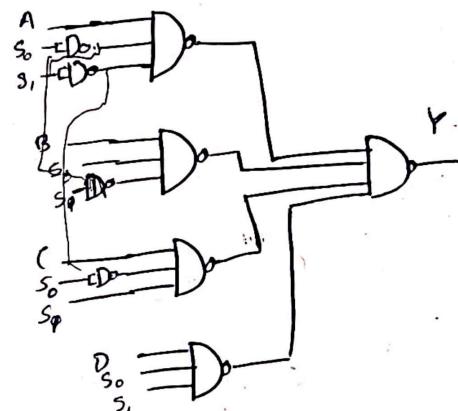
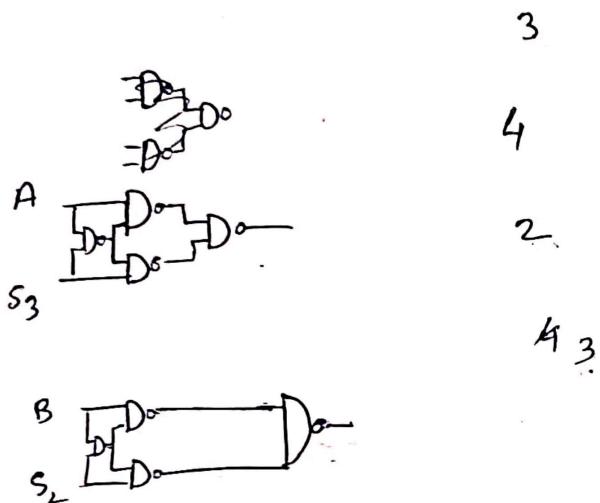
TH-181 \Rightarrow bit sliceable ALU

ADD, SUB, XOR, AND

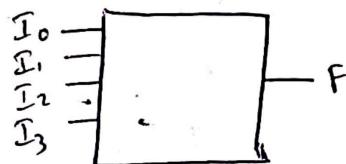




MATRIX
DECODE → LE
Matrix → LC



43 LE → Logic Element



$\frac{xc7}{\downarrow}$ $\frac{35t}{635 \text{ kcells}}$ CGP 238
 - extension cell within ↳ package tow
 32 kLUT SMD - 238 pin
 Matrix 7 speed programming.

→ 100 MHz

Carry look ahead adder.

$\frac{n(n+1)}{2}$ AND, n OR gate

ALU - 1 bit

$a_{in} = \begin{bmatrix} + \\ - \end{bmatrix}$

$b_{in} = 1 \text{ or } 0$

carry-in

result-out

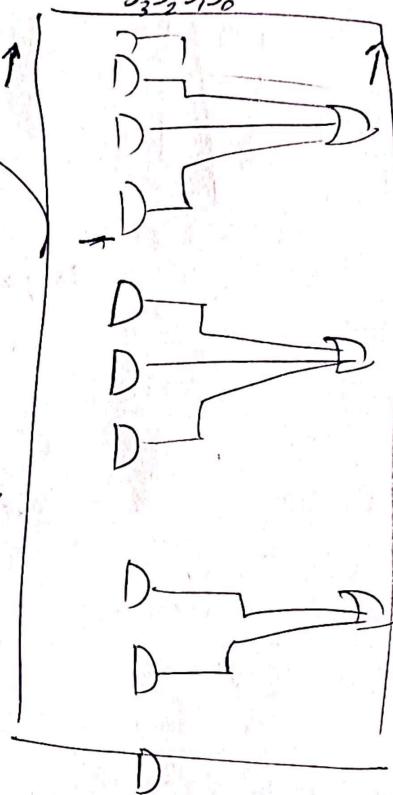
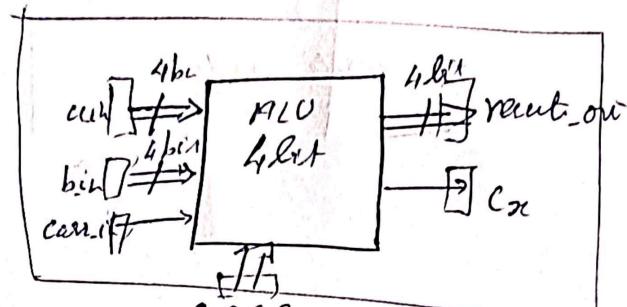
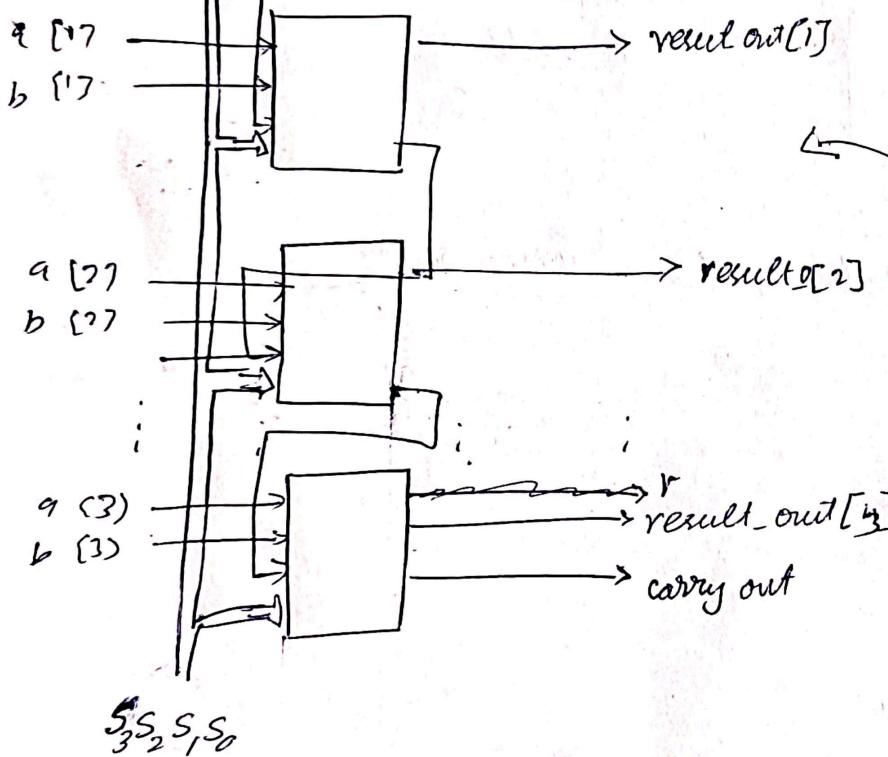
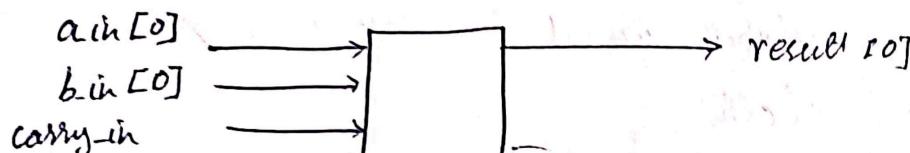
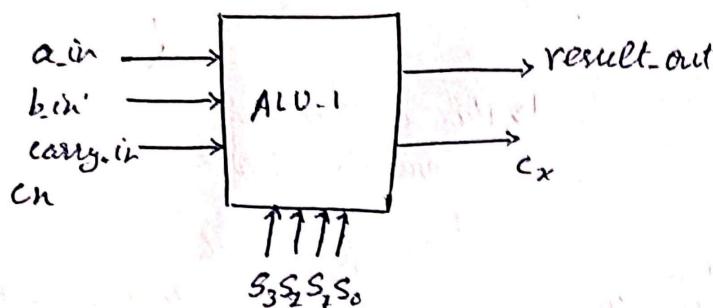
carry-out

1
2
3
4

$a_{in} = 1011$

$b_{in} = 1101$

$c_{in} = 000$



Advantage

→ Less time

Disadvantage

→ Increase complexity

→ adds more gates to design

→ n-bits

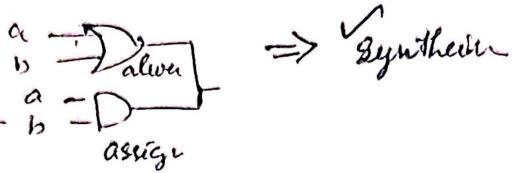
$$\frac{n(n+1)}{2} \text{ AND}$$

n - OR

1×2
 $1 + 2 + 3 -$

Run Xstler

- ↳ avoid multidriver.
- ↳ no-connected nets
- ↳ latch after



↳ Run synthesis.

Constraint wizard

↳ before finish

view timing constraint

Create timing summary report

generates.

↳ Clock wizard.

clk

→ 200 MHz

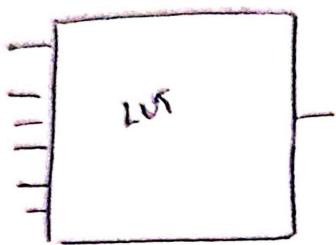
→ select signals

→ initial condition.

Finish.

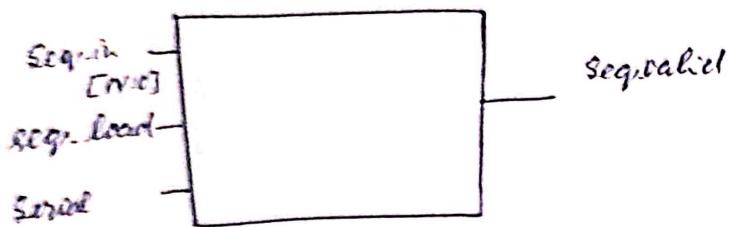
⇒ generate report ⇒ Timing report

use of race \Rightarrow optimize for ALU

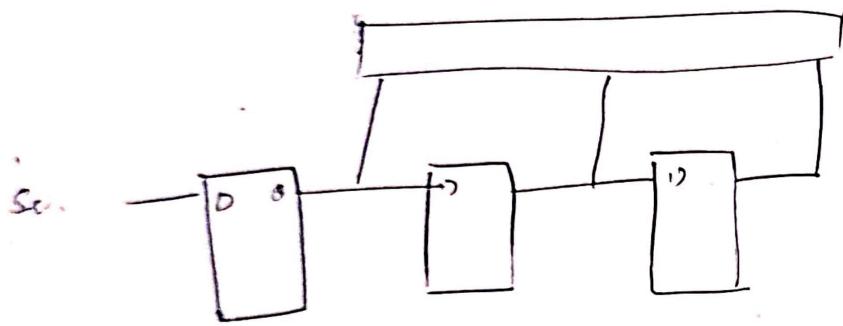


LFFR - Linear feed forward register.

8.84

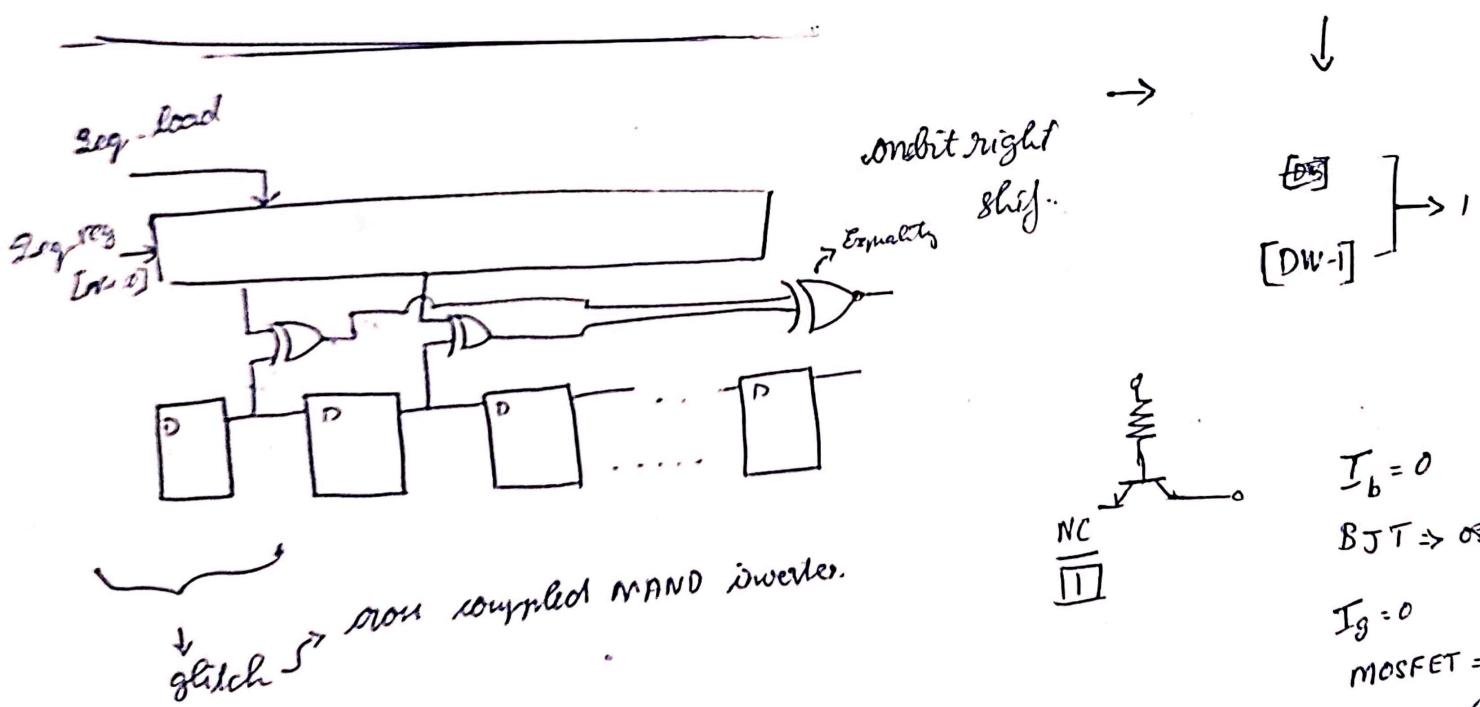


$5A5A5A$
6x4



0101 1010 $\times 3$
1-
0-

~~X~~ $d_2 d_2 d_2$
 $\underline{0}$ → 1



(a) transitional rate $\frac{Q_1 Q_2}{0 \rightarrow 1}$
 \Rightarrow floating NO internal PULLUP
 PULL-DOWN } register gate
 in FPGAs

MOSFET \Rightarrow can be used to implement
 but, generates impulse response. \Rightarrow

VHDL \Rightarrow state variable.
 Verilog \Rightarrow 4 state variable \Rightarrow that cannot account for
 floating state in a BJT
 \hookrightarrow commercial design they use MOSFET so does not
 have to account for.

Motability avoided by reguring o/p

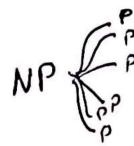
BCD \Rightarrow 7 segment

$$\begin{matrix} A \\ F \mid \frac{a}{D} \mid B \\ E \mid \frac{c}{D} \mid C \end{matrix}$$

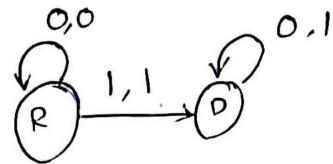
| | B ₃ | B ₂ | B ₁ | B ₀ | A | B | C | D | E | F | G |
|---|----------------|----------------|----------------|----------------|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 2 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 6 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

Automata Theory -

- Non-probabilistic problem
- Probabilistic problem
- ↳ finite step.



$\frac{101}{\downarrow}$ RESET $\rightarrow 1 \text{ DET_1}$
 10 RESET



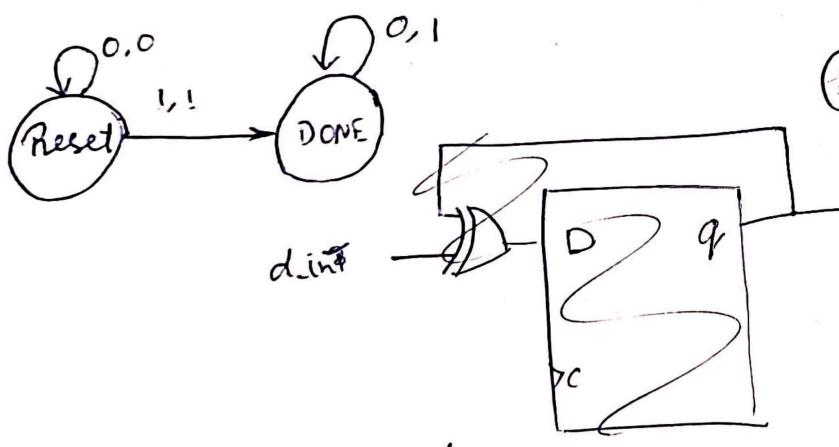
DET_1 $\rightarrow 0 \text{ DET_10}$
 $| \quad | \text{ DET_1}$

DET_10 $\rightarrow 1 \text{ DONE}$
 10 RESET

DONE $\rightarrow 1 \text{ DET_1}$
 10 DET_10

$d_{in} = 10110110111\dots$
 $\downarrow \downarrow \downarrow \downarrow \downarrow$
 $q_f = 1101101$

$\begin{matrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$
10

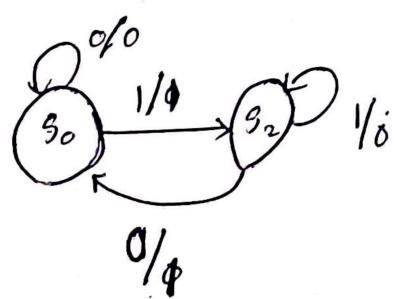


$d - 0 \quad 1 \quad 1 \quad 0$
 $q_f - \underline{\underline{0}} \underline{\underline{1}} \underline{\underline{1}} 0 \rightarrow$

$\begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix}$

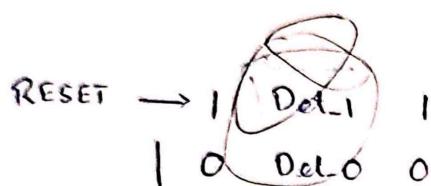
if (d_{in})
 $q_f = \sim q_{out}$

else
 $q_f = q_{out}$



RESET $\rightarrow 1 \text{ DONE } 1$
 $10 \text{ RESET } 0$
DONE $\rightarrow 1 \text{ DONE } 0$
 $10 \text{ RESET } 1$

odd is a even 'o's

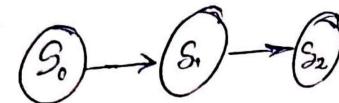


Det-1 → 0 Det-0 0

1 1 Det-11 0

1
↓
01

100
010
001



Det-0 → 1 Det-01 1

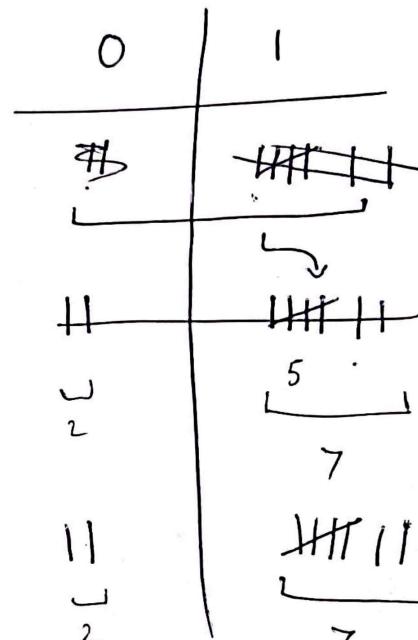
1 0 Det-00 1

RESET → 1 DONE , 1
 1 0 E1-00 , 0

E1-00 → 1 01-00 , 0
 1 0 RESET , 0

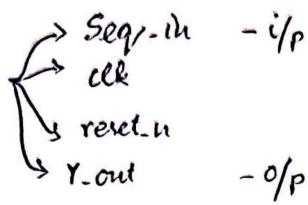
01-00 → 1 E1-00 , 0
 1 0 DONE , 1

DONE → 1 RESET , 0
 1 0 odd1_odd0 , 0



→ Odd & Even.

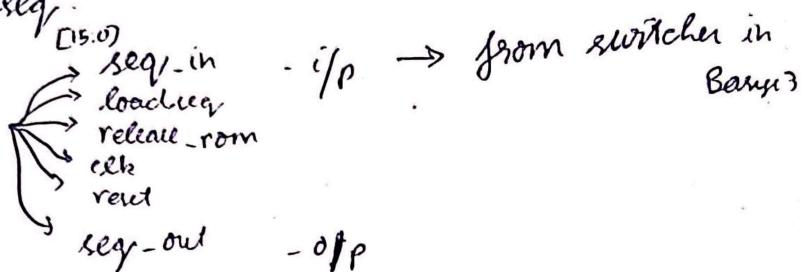
⇒ detect odd, even SV



TB ✓

→ ROM to store & release seq.

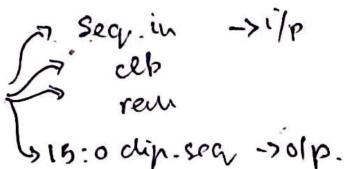
⇒ input.rom.SV



from switcher in
Bank 3

→ counter & display.

⇒ ~~test~~ counter.display.SV



06-11-2024
Design Patterns in System Verilog HDL

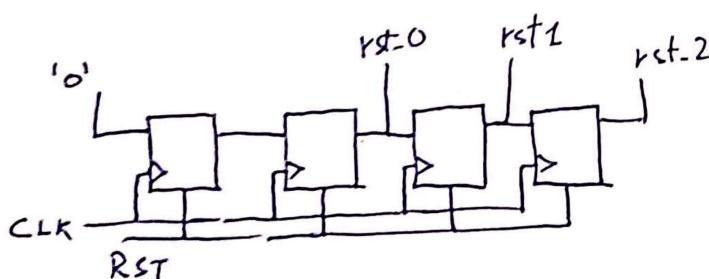
- S. clean SRAM

- reset

aktion 7

↳ done by a RESET-synchronous
of time period 100ms

↓
Test Bench reset

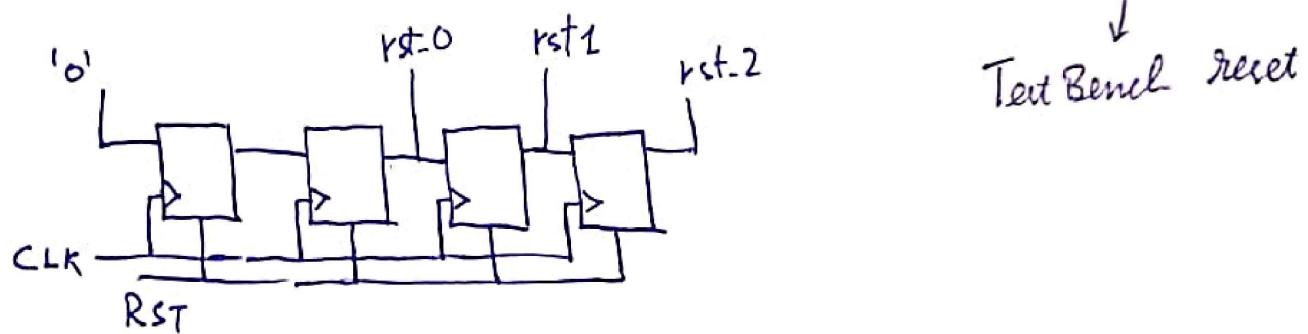


2. clear SRAM

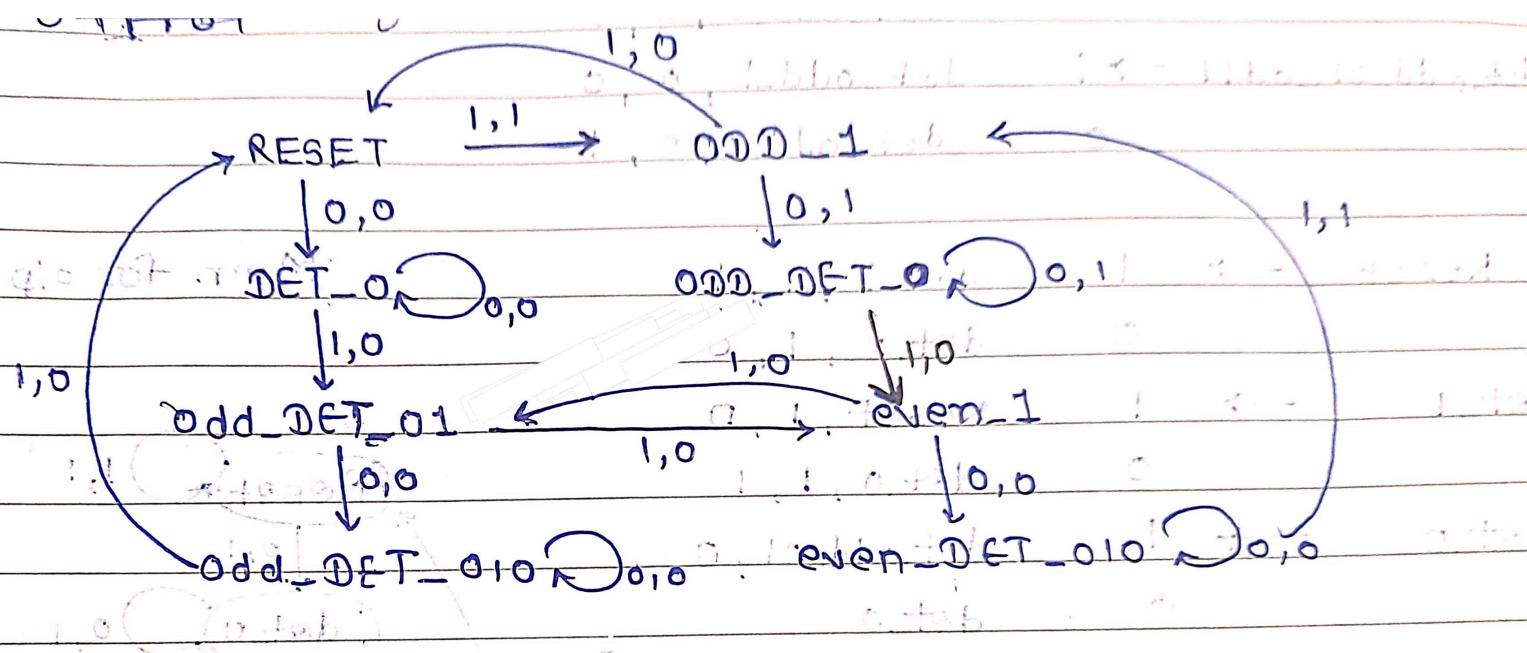
- reset

Section 7

↳ done by a RESET synchronous
of time period 100ms



Sequence detector odd 1's and even 01 toggles



seq - 010100110100001011100
 exp - 000000000011110010000000
 o/p - 000000000011110010100000

Mute → Toggle

16ms

20ms

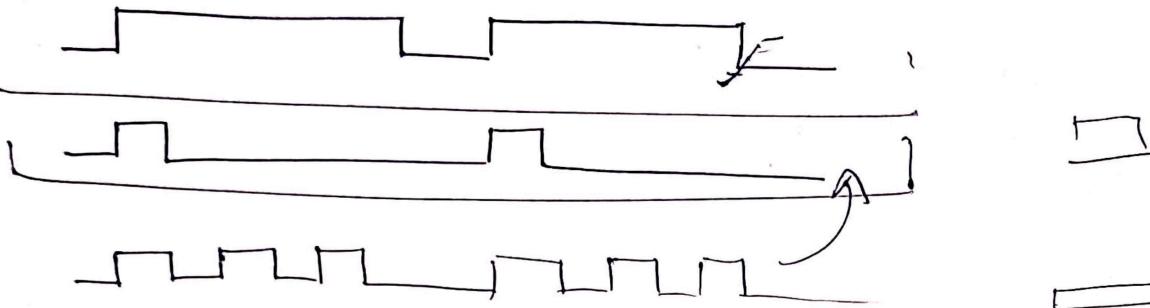
62.5 kHz

Volume UP → press & hold

$\frac{1}{20} \times 10^{-3}$

50Hz

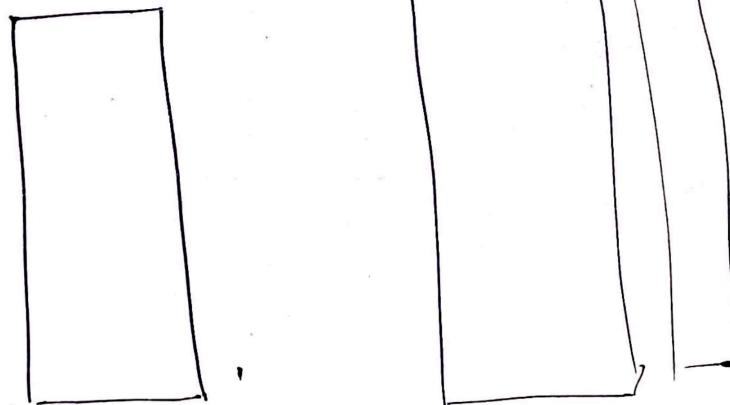
Phy



↑ ↓ x MUTE

↑ + - ↓

MUTE



PS/2 → OS/2
PS/2 → DB/2

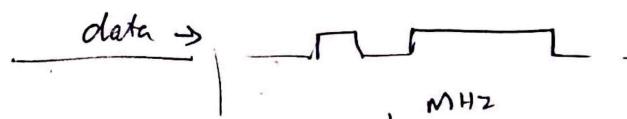
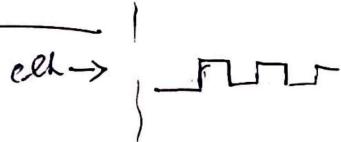
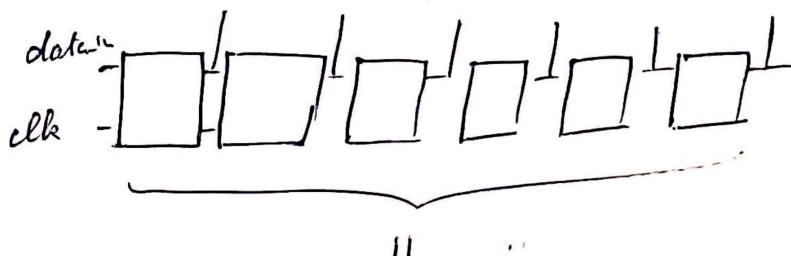
accept info from synchronous system & display over 7-segment digit.

→ Vcc and
5 - clk 1 - data

↓

shift - register & latch

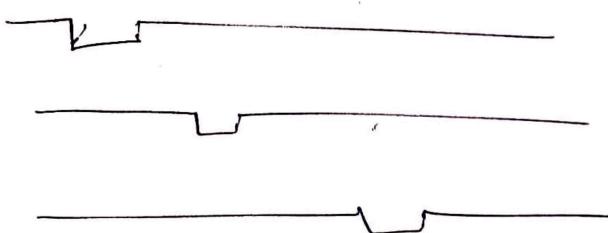
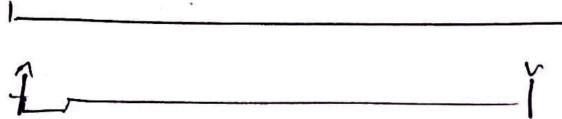
Data-width → 11 bits.

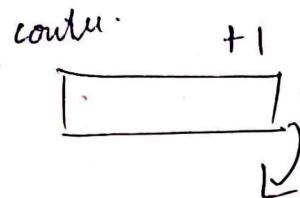


10⁻³

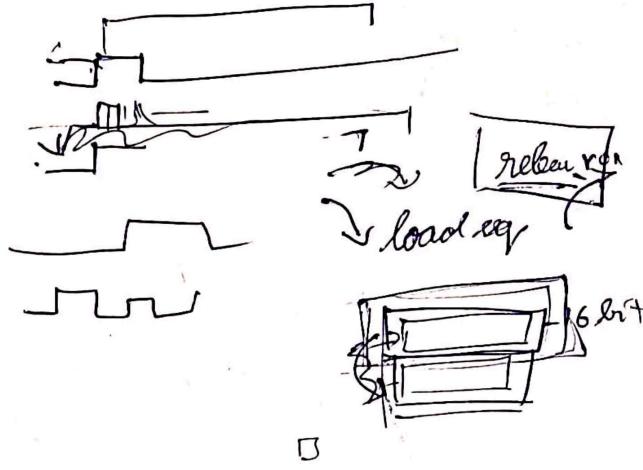
1kHz →
4kHz

refreshing frequen-





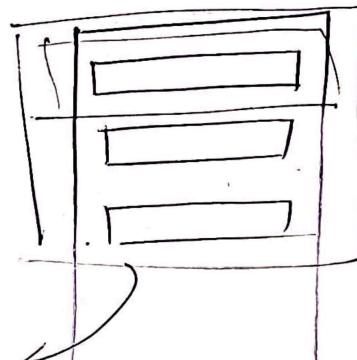
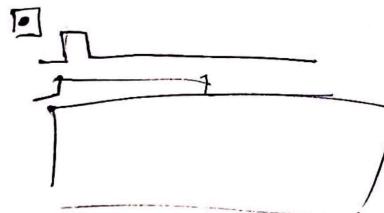
To 7-seg display.



→ store sequence in ROM

→ on run, fetch the sequence & detect segment.

```
for (i=0 ; i <= 3 ; i++)  
    rom mem[i] <= 0;
```



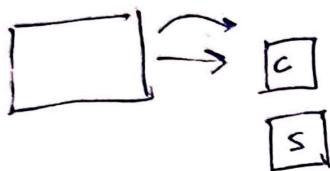
[15:0] ~~mem~~ ROM [3:0]
~~[0:3]~~
~~[0:3]~~

ptr = 0

→ load seq
 -> n[15] ptr < B4
 ROM [0] <= 0
 n[15] = +1
 +1201

ARM → Bus ~ 1.
Memory ~ 2.
UVM ~ 3.
→ Master, Slave, ...
AXI, AXB, ...

Out of order
done above coding
UVM based
coverage from
bsm

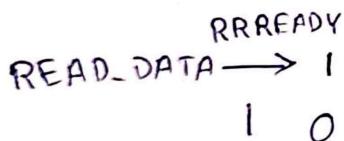
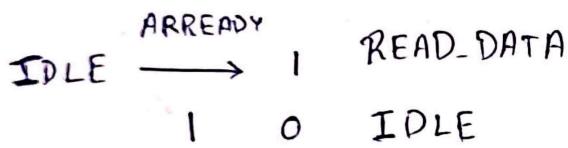


```
if (sig[i]) begin
    mem[i] <= in-data;
end
```

mem[0] 
mem[i] 

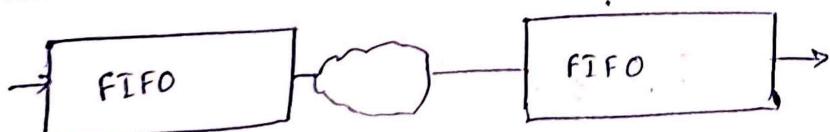


Read Master



13-11-2024

Single clk pipeline



2

- ✓ works without metastability
- ✓ wider frequency range

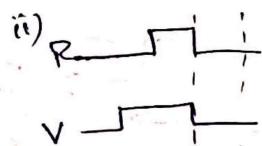
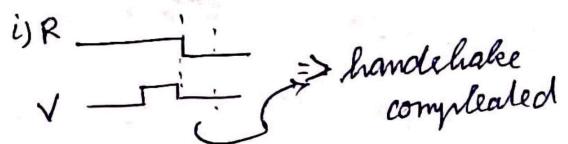
AMBA - AXI - A Lite

READY, VALID \Rightarrow Common for different modes.

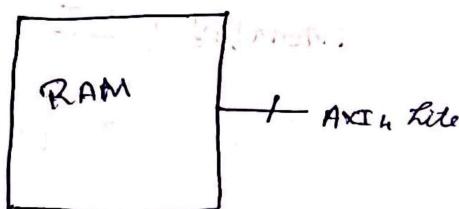


High \rightarrow channel should have valid data

Handshake



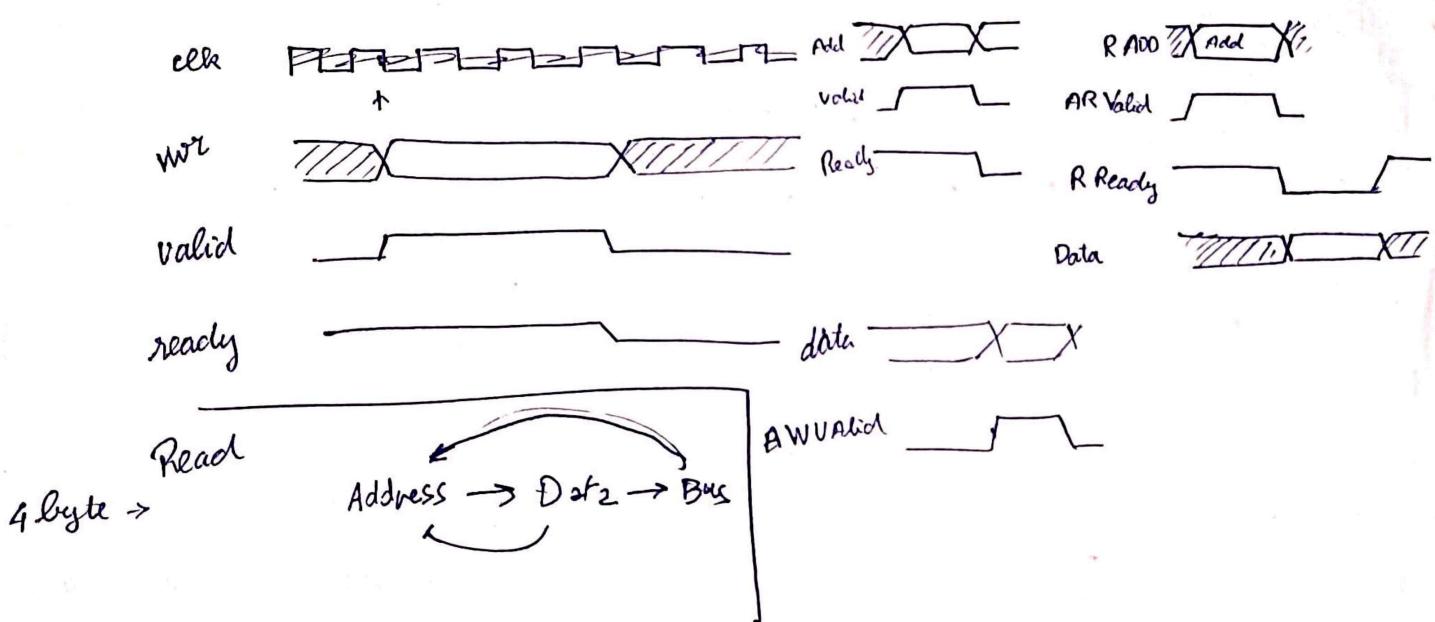
- Variable width verification RAM, with AXI 4 subordinate



address Channel in AXI₄

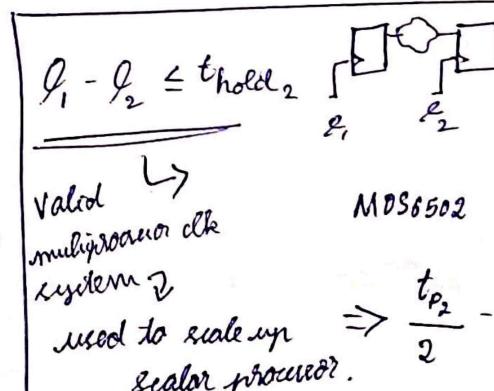
READ

WRITE
Master \rightarrow



Address Handshake

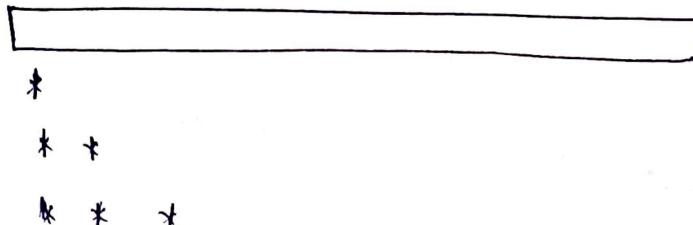
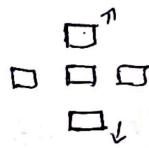
Data Handshake



\rightarrow Implementation of overclocking. AMD alternatively

\rightarrow single clk pipeline

\rightarrow higher throughput at same clock frequency



~~IDLE~~

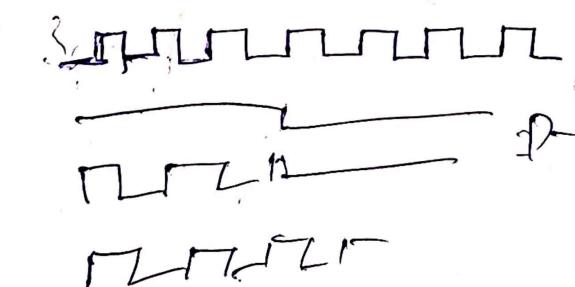
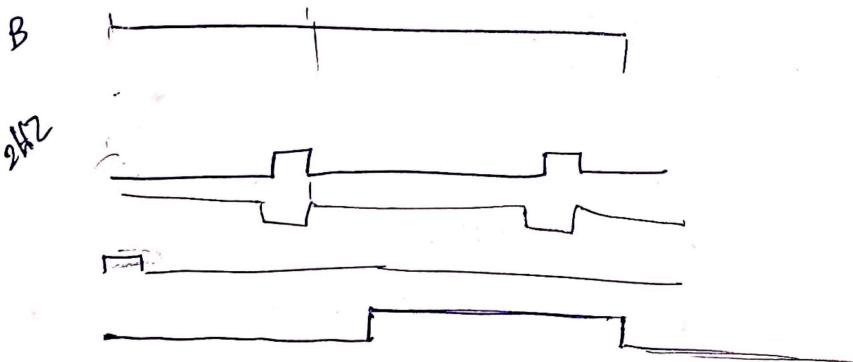
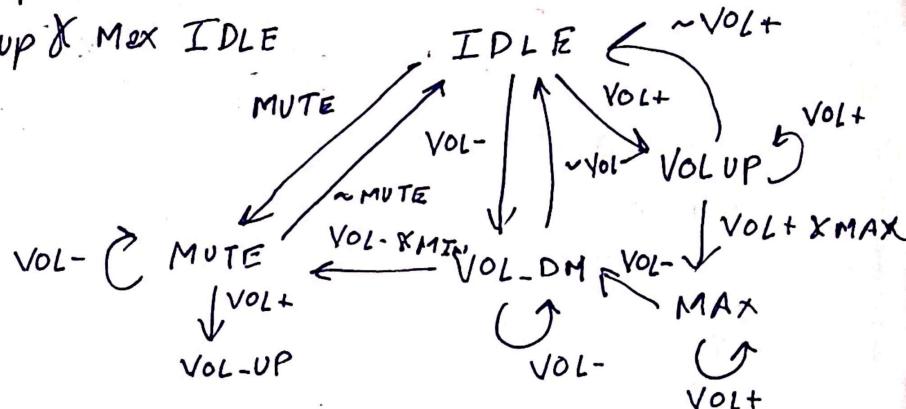
MUTE → mute MUTE
↓ ~mute IDLE

| up VOL-UP MUTE → ~mute & mute clr IDLE
| down MUTE

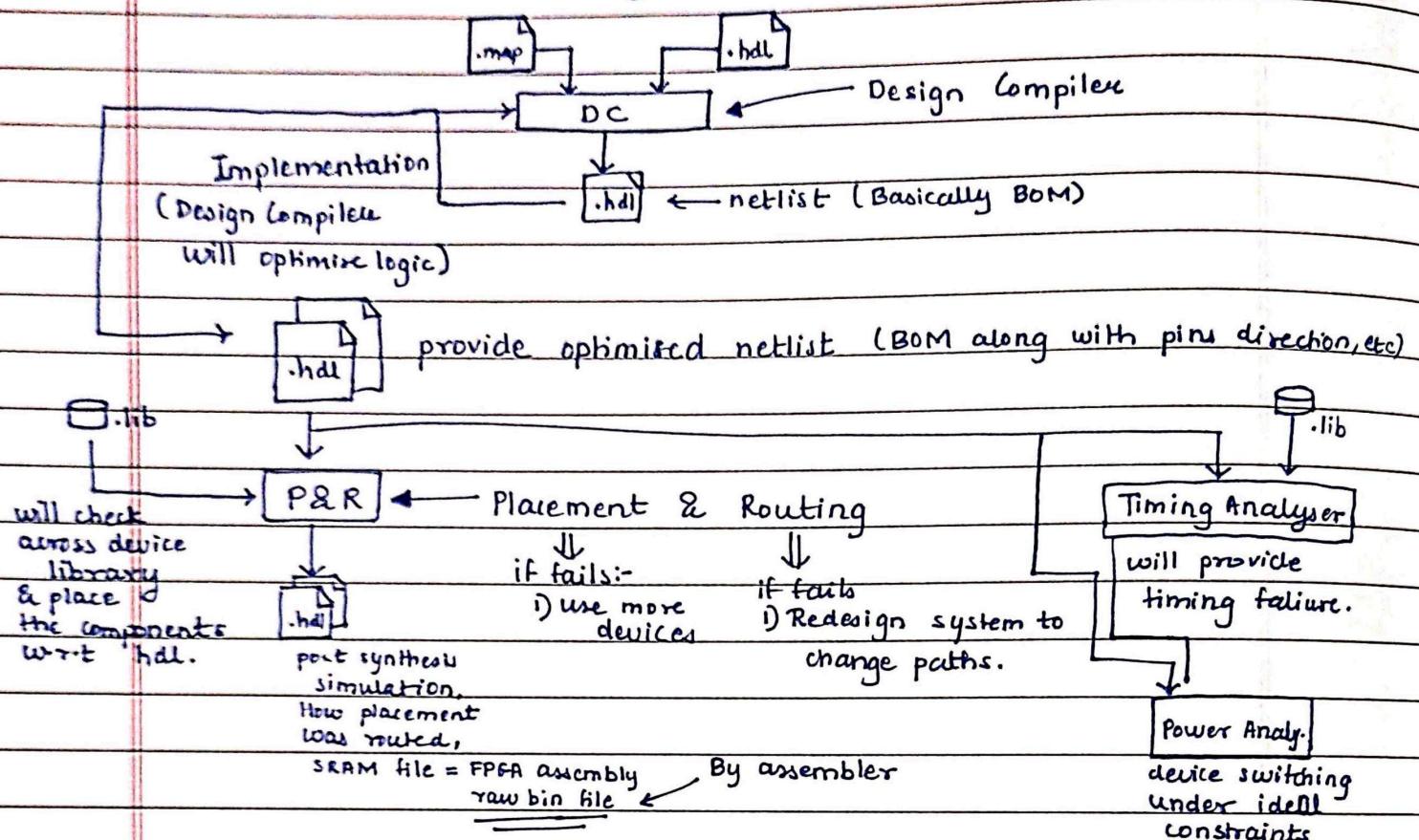
VOL-UP → ~up IDLE

| up VOL-UP

| up & Max IDLE



* Reporting (Vivado) :- Vivado generates report at every stage except Linting & Elaboration.



6 types of reporting. & 3 more:-
 1) utilization - No. of reg, buf, nets and all other used components
 2) DRC :- If there exists any violations in designs.
 3) Methodology :- parameters or how design was generated.

Floorplanning & IO planning.

TOP right ~~right~~ \Rightarrow SRAM.

$X_0 Y_1$ $X_0 Y_0$ \Rightarrow UIK regions

very good!!!

bitstream = from SRAM file to PPGA SRAM (Raw bin to Flash mem)

Mstream MCS = more reliable than raw bin.
 memory calibration system.

Low Power VLSI design

DFT, xv.log, xsim.

Design constraint.

Timing analyser, FPGA analyser, simulator.

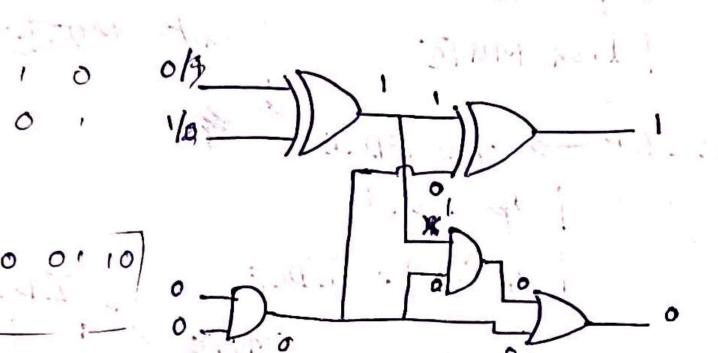
Method

① → out of control synthesis.

② → Block design.

Design for testability

Scan flop



A B C D.

0 1 0 0

1 0 0 0

0 1 0 1

1 0 0 1

0 1 1 0

1 0 1 0

Block Designer.

AKT BRAM controller

Block mem generator

Altera
Lookahead

• LE vs LUT

• memory

Bulk memory vs
Throughput

Latency

Xilinx

LUT → config time

distributed vs area

DRAM, between region

high density

iCE40

small form
small power

ECP5

Small factor
High perfom