

CMOS Logic

Carsten Wulff, carsten@wulff.no

I. CMOS LOGIC

Status: 0.3

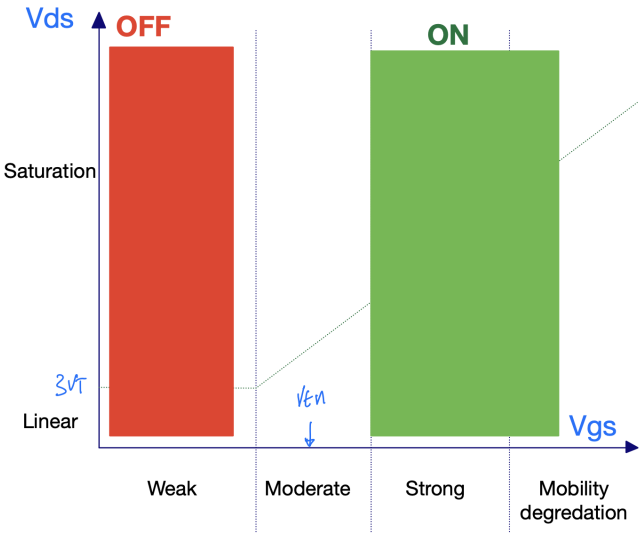
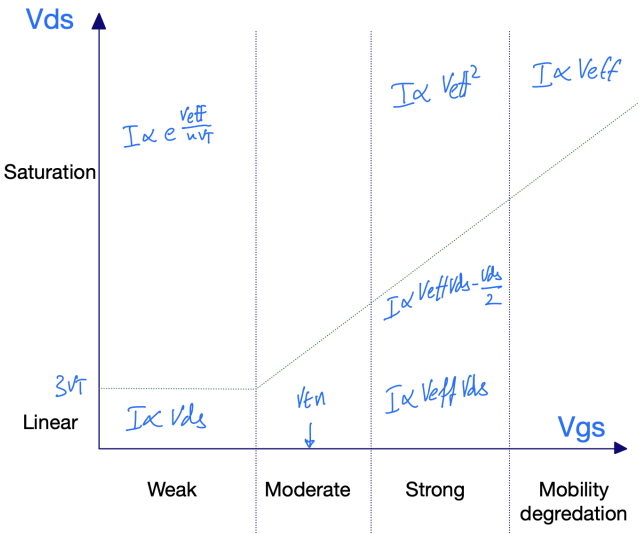
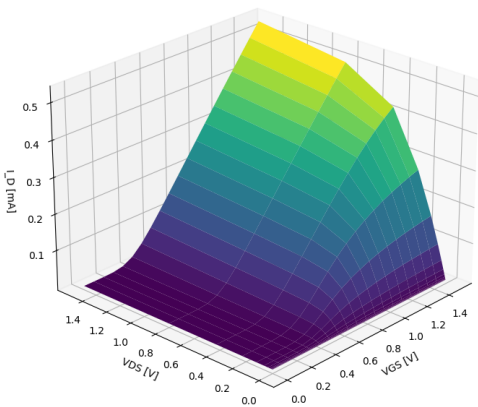
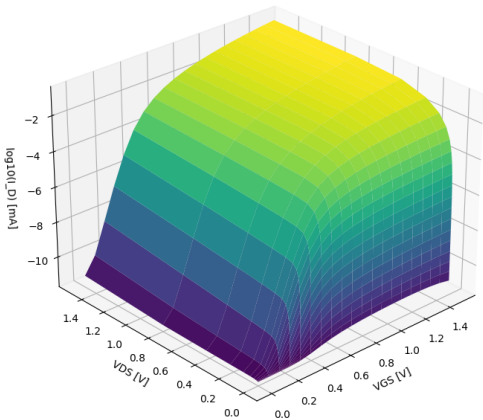
II. ANALOG TRANSISTOR TO DIGITAL TRANSISTOR
NMOS current ($W = 0.4\mu$ $L=0.15\mu$) as a function of

$$V_{GS}$$

and

$$V_{DS}$$

dicex/lectures/l13/mos.py



Gate	NMOS	PMOS
VDD	ON	OFF
VDD -> VSS	X	X
VSS -> VDD	X	X
VSS	OFF	ON

Gate	NMOS	PMOS
1	ON	OFF
1 -> 0	X	X
0 -> 1	X	X
0	OFF	ON

III. CMOS STATIC LOGIC ASSUMPTIONS

NMOS source is connected to low potential

$$V_{GS} > V_{TH}$$

when

IV. DON'T BREAK RULES UNLESS YOU KNOW EXACTLY WHY IT WILL BE OK

$$V_G = V_{DD}$$

V. LOGIC CELLS

PMOS source is connected to high potential

$$V_{GS} < V_{TH}$$

when

$$V_G = 0$$

A	B	$\overline{AB}$	$\overline{A+B}$	$\overline{AB}$	$\overline{A+B}$
0	0	1	1	0	0
0	1	1	0	0	1
1	0	1	0	0	1
1	1	0	0	1	1

$$\overline{AB} = \overline{A+B} \quad \leftarrow \text{DM}$$

$$\overline{A+B} = \overline{A} \overline{B} \quad \leftarrow \text{DM}$$

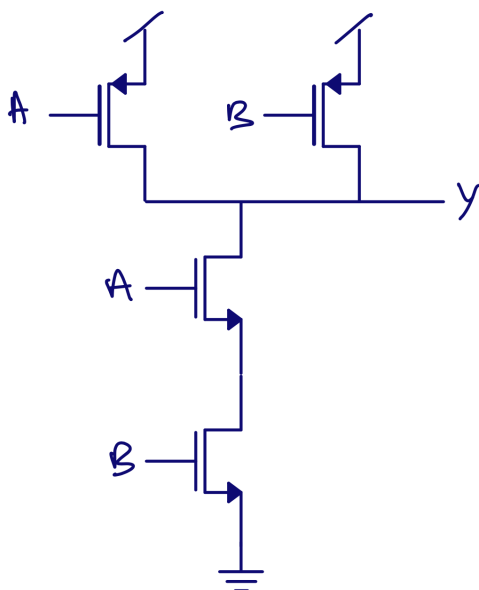
$$AB = \overline{\overline{A+B}}$$

$$A+B = \overline{\overline{A} \overline{B}}$$

$\overline{A}$	$\overline{B}$	$\overline{A+B}$	$\overline{AB}$	$\overline{A+B}$	$\overline{AB}$
1	1	1	1	0	0
1	0	1	0	0	1
0	1	1	0	0	1
0	0	0	0	1	1

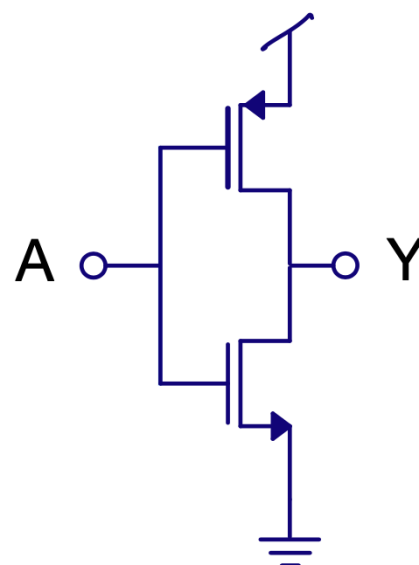
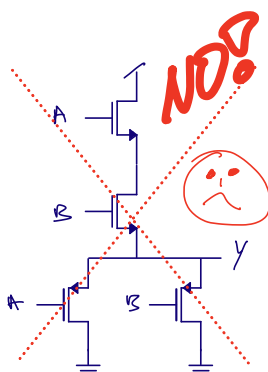
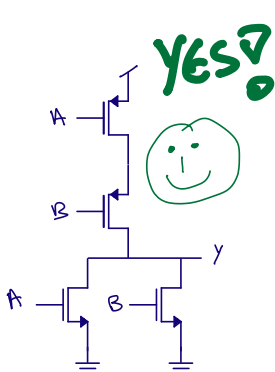
$$\overline{AB} \neq \overline{A} \overline{B}$$

$$\overline{A+B} \neq \overline{\overline{A} \overline{B}}$$



A. CMOS static logic is inverting

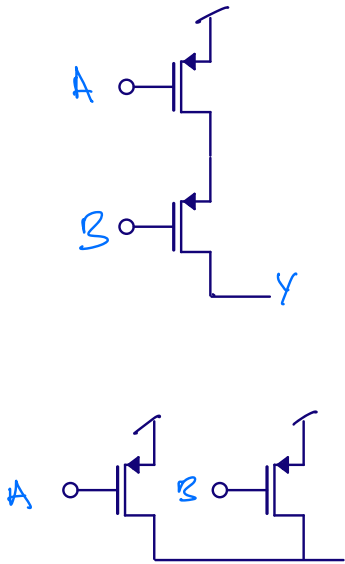
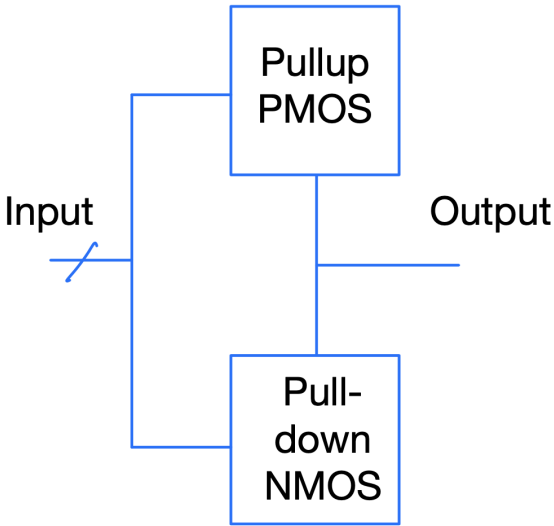
A	Y
1	0
0	1



PD \ PU	OFF	ON
	OFF	ON
OFF	Z	1
ON	0	X

PD = Pull-down PU = Pull-up

```
logic => [0,1,Z,X];
```



[.table-separator: #000000, stroke-width(1)] [.table: margin(8)]

Pull-down series

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

Pull-down paralell

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

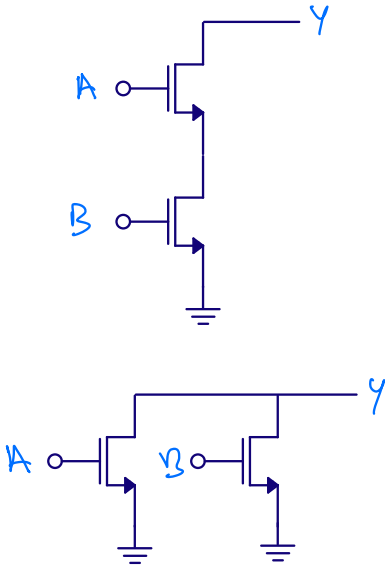
[.table-separator: #000000, stroke-width(1)] [.table: margin(8)]

Pull-up series

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

Pull-up paralell

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



B. Rules for inverting logic

Pull-up OR

=>

PMOS in series

=>

POS AND

⇒

PMOS in paralell

⇒

PAP

Pull-down OR

⇒

NMOS in paralell

⇒

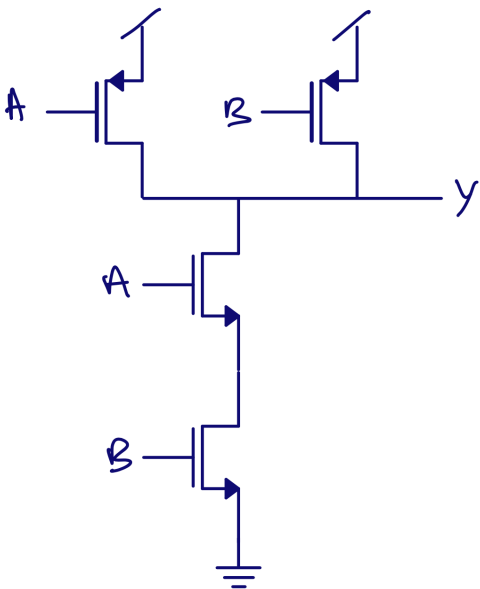
NOP AND

⇒

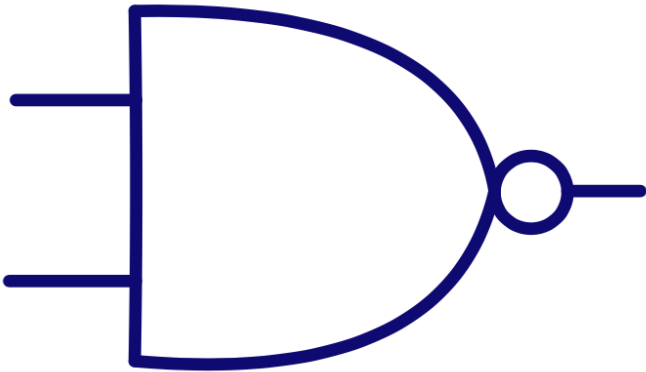
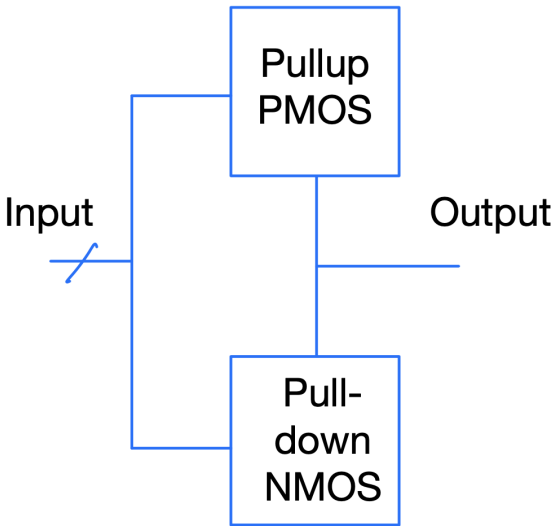
NMOS in series

⇒

NAS



A	B	NOT(A AND B)
0	0	1
0	1	1
1	0	1
1	1	0



[.table-separator: #000000, stroke-width(1)] [.table: margin(8)]

[.table-separator: #000000, stroke-width(1)] [.table: margin(8)]

$Y = \overline{A + B} = \text{NOT} (A \text{ OR } B)$

$Y = \overline{AB} = \text{NOT} (A \text{ AND } B)$

AND PU

⇒

PMOS in paralell PD

⇒

NMOS in series

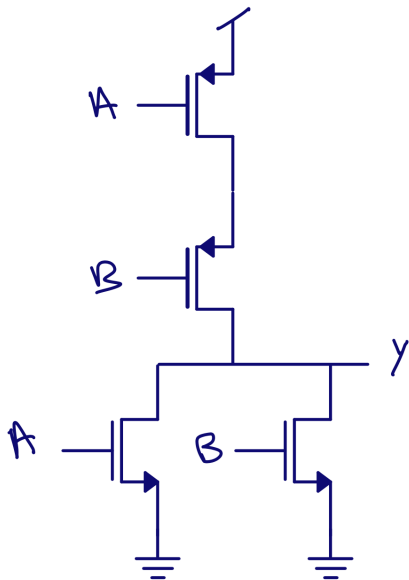
OR PU

⇒

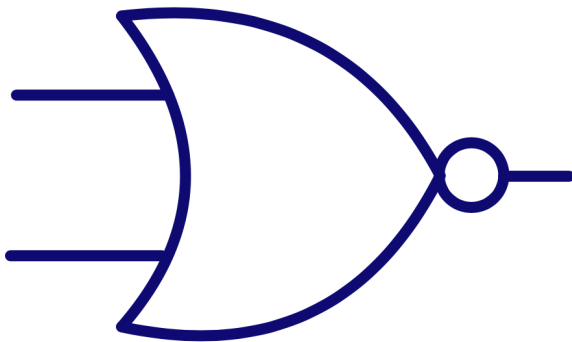
PMOS in series PD

⇒

NMOS in paralell



A	B	NOT(A OR B)
0	0	1
0	1	0
1	0	0
1	1	0



VI. SR-LATCH

Use boolean expressions to figure out how gates work.

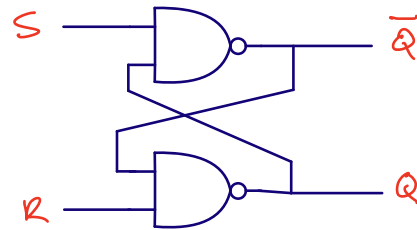
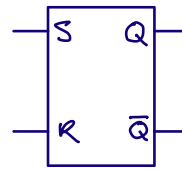
Remember De-Morgan

$$\overline{AB} = \overline{A} + \overline{B}$$

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

$$Q = \overline{R\overline{Q}} = \overline{R} + \overline{\overline{Q}} = \overline{R} + Q$$

$$\overline{Q} = \overline{S\overline{Q}} = \overline{S} + \overline{\overline{Q}} = \overline{S} + Q$$



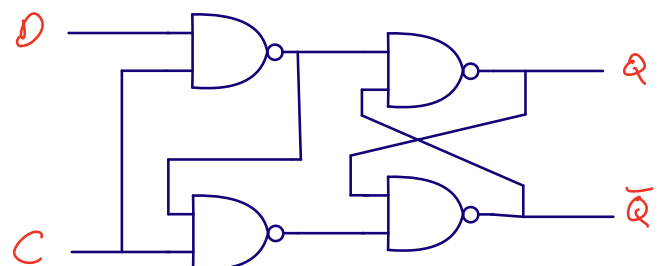
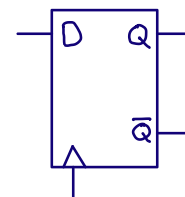
$$Q = \overline{R} + Q$$

$$\overline{Q} = \overline{S} + \overline{Q}$$

S	R	Q	~Q
0	0	X	X
0	1	0	1
1	0	1	0
1	1	Q	~Q

VII. D-LATCH (16 TRANSISTORS)

C	D	Q	~Q
0	X	Q	~Q
1	0	0	1
1	1	1	0



VIII. OTHER LOGIC CELLS

What about

$Y = AB$

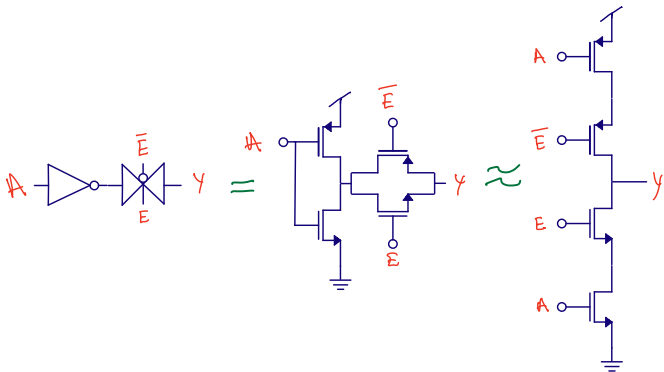
and

$Y = A + B$

?

$Y = AB = \overline{\overline{AB}}$

$Y = A \text{ AND } B = \text{NOT}(\text{NOT}(A \text{ AND } B))$



[.table-separator: #000000, stroke-width(1)] [.table: margin(8)]

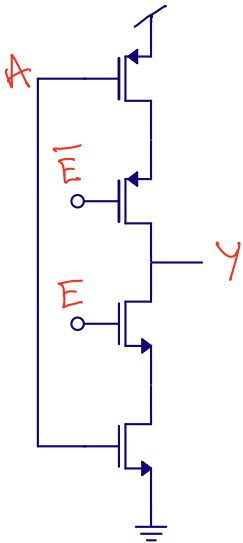
X. TRISTATE INVERTER

$Y = A+B = \overline{\overline{A+B}}$

$Y = A \text{ OR } B = \text{NOT}(\text{NOT}(A \text{ OR } B))$



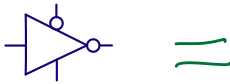
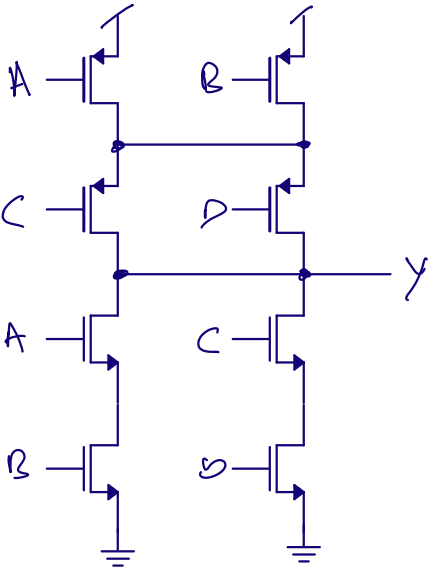
E	A	Y
0	0	Z
0	1	Z
1	0	1
1	1	0



IX. AOI22: AND OR INVERT

$Y = \text{NOT}(A \text{ AND } B \text{ OR } C \text{ AND } D)$

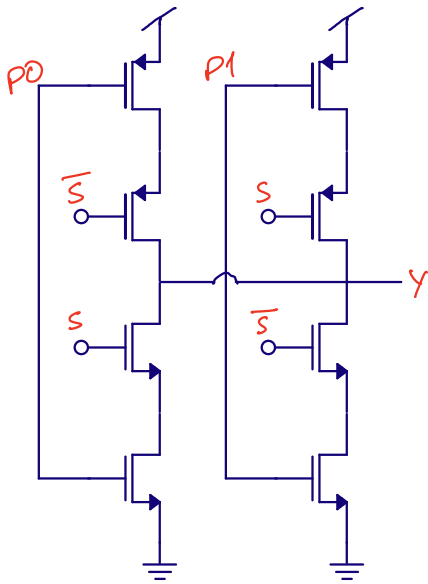
$Y = \overline{AB + CD}$



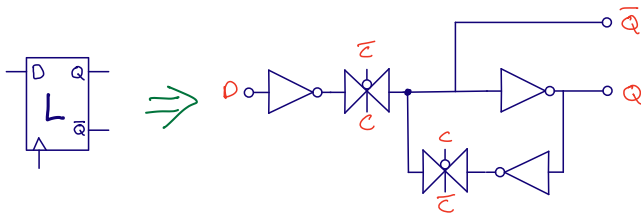
[.table-separator: #000000, stroke-width(1)] [.table: margin(8)]

XI. MUX

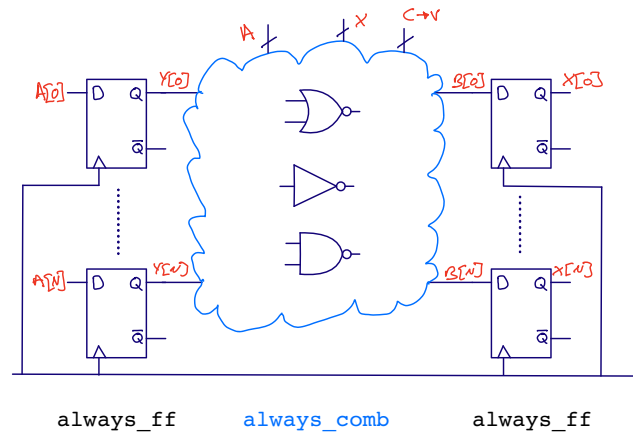
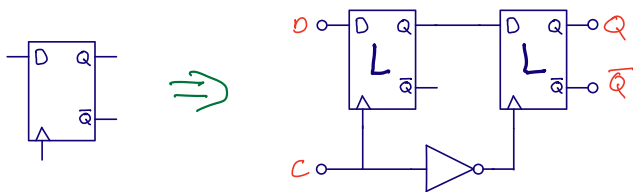
S	Y
0	NOT(P1)
0	NOT(P1)
1	NOT(P0)
1	NOT(P0)



D-Latch (12 transistors)



D-Flip Flop (< 26 transistors)



always_ff

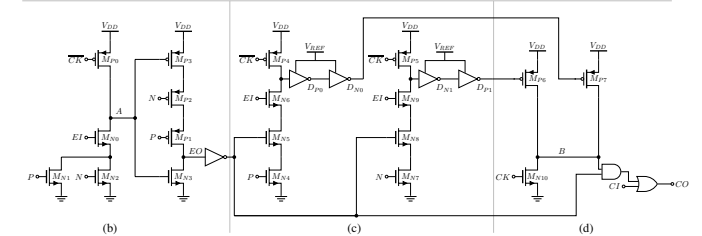
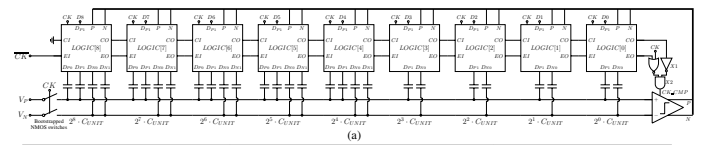
always_comb

always_ff

XII. THERE ARE OTHER TYPES OF LOGIC

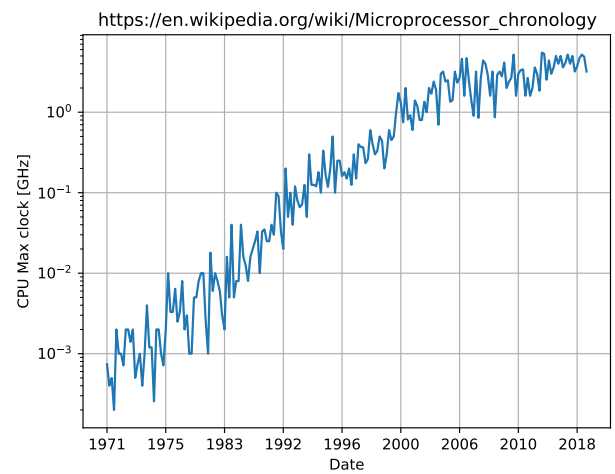
- True single phase clock (TSPC) logic
- Pass transistor logic
- Transmission gate logic
- Differential logic
- Dynamic logic

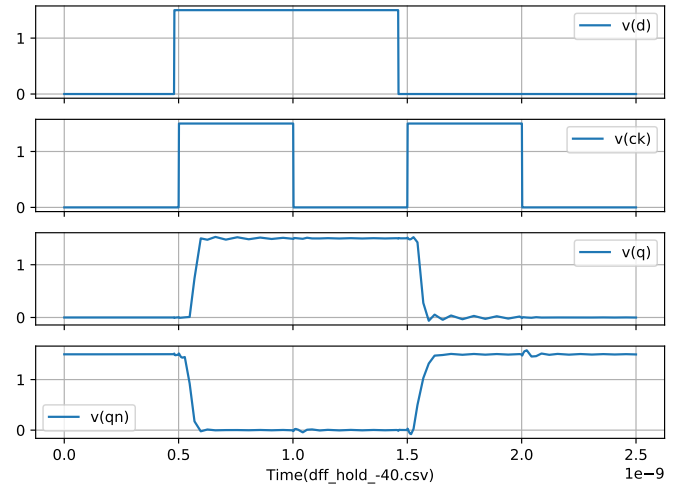
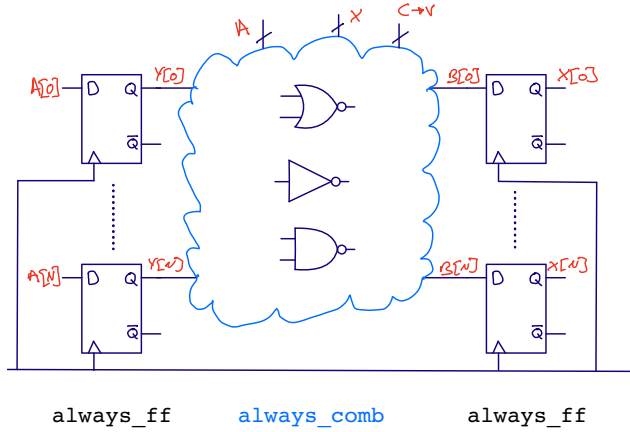
Consider other types of logic “rule breaking”, so you should know why you need it.



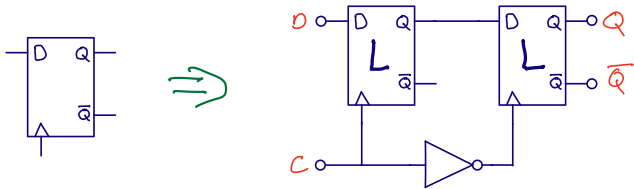
Dynamic logic => A Compiled 9-bit 20-MS/s 3.5-fJ/conv.step SAR ADC in 28-nm FDSOI for Bluetooth Low Energy Receivers

XIII. SPEED





XIV. FLIP-FLOPS AND SPEED



XV. TIMING ANALYSIS

Analyze arrival times of all nodes in a combinatorial circuit

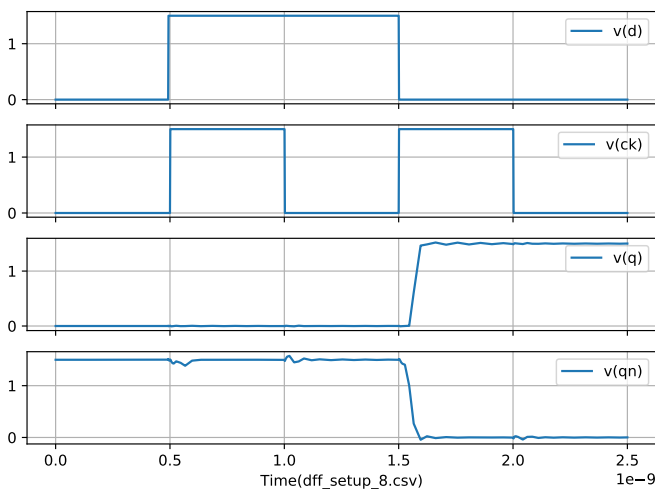
$$arrival_j = \max_{j \in fanin(i)} arrival_j + t_{pd_i} \Rightarrow a_j = \max_{j \in fanin(i)} a_j + t_{pd_i}$$

$$slack_i = required_i - arrival_j$$

dicex/lib/SUN_TR_GF130N.spi:

```
.SUBCKT DFRNQN1_CV D CK RN Q QN AVDD AVSS
XA0 AVDD AVSS TAPCELLB_CV
XA1 CK RN CKN AVDD AVSS NDX1_CV
XA2 CKN CKB AVDD AVSS IVX1_CV
XA3 D CKN CKB A0 AVDD AVSS IVTRIX1_CV
XA4 A1 CKB CKN A0 AVDD AVSS IVTRIX1_CV
XA5 A0 A1 AVDD AVSS IVX1_CV
XA6 A1 CKB CKN QN AVDD AVSS IVTRIX1_CV
XA7 Q CKN CKB RN QN AVDD AVSS NDIRIX1_CV
XA8 QN Q AVDD AVSS IVX1_CV
.ENDS
```

Setup time: How long before clk does the data need to change



Hold time: How long after clk can the data change

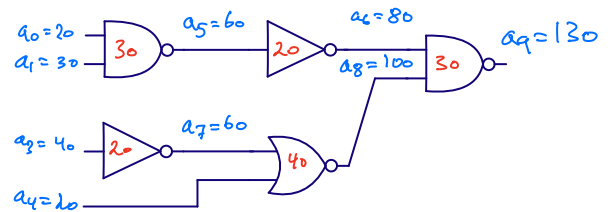
Positive slack (over PVT¹)

⇒

Timing is OK Negative slack (over PVT²)

⇒

Timing is not OK



XVI. TIMING ANALYSIS TOOLS

Commercial [Cadence Tempus](#)

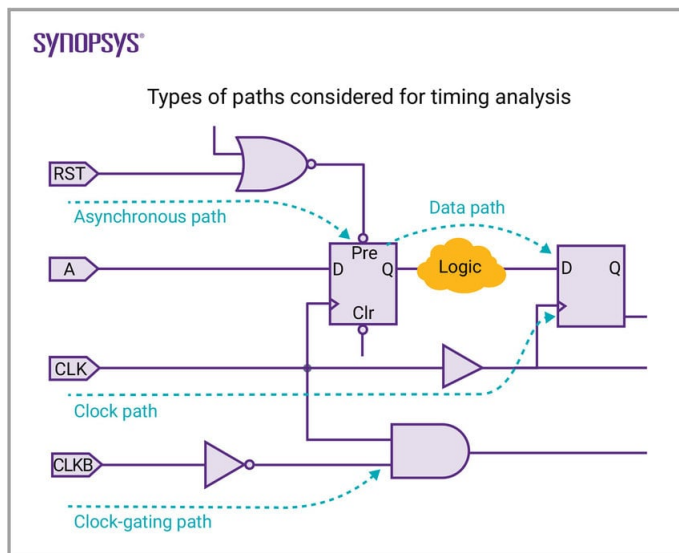
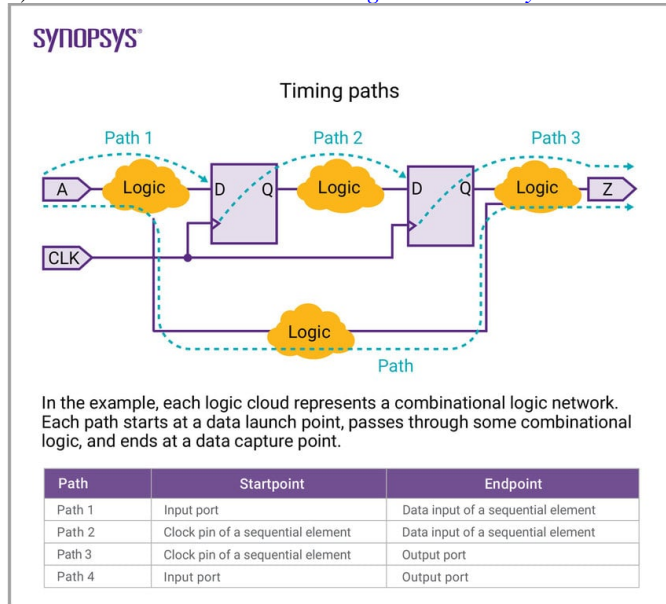
[Synopsys PrimeTime](#)

Free [OpenTimer](#)

¹Often called *clock gating*

²Often called *clock gating*

1) What is timing analysis:



2) *What do the tools need?*: Input and output delay paths as a function of input transition time and capacitive load, setup and hold time.

osu018_stdcells.lib

```
cell (INVX1) {
  cell_footprint : inv;
  area : 16;
  cell_leakage_power : 0.0221741;
  pin(A) {
    direction : input;
    capacitance : 0.00932456;
    rise_capacitance : 0.00932196;
    fall_capacitance : 0.00932456;
  }
  pin(Y) {
    direction : output;
    capacitance : 0;
    rise_capacitance : 0;
    fall_capacitance : 0;
    max_capacitance : 0.503808;
    function : "(!A)";
    timing() {
```

```
related_pin : "A";
timing_sense : negative_unate;
cell_fall(delay_template_5x5) {
  index_1 ("0.005, 0.0125, 0.025, 0.075, 0.15");
  index_2 ("0.06, 0.18, 0.42, 0.6, 1.2");
  values ( \
    "0.030906, 0.037434, 0.038584, 0.039088, 0.030318", \
    "0.04464, 0.057551, 0.073142, 0.077841, 0.081003", \
    "0.064368, 0.091076, 0.11557, 0.126352, 0.144944", \
    "0.139135, 0.174422, 0.232659, 0.261317, 0.321043", \
    "0.249412, 0.28434, 0.357694, 0.406534, 0.51187");
}

fall_transition(delay_template_5x5) {
  index_1 ("0.005, 0.0125, 0.025, 0.075, 0.15");
  index_2 ("0.06, 0.18, 0.42, 0.6, 1.2");
  values ( \
    "0.032269, 0.0648, 0.087, 0.1032, 0.1476", \
    "0.036025, 0.0726, 0.1044, 0.1236, 0.183", \
    "0.06, 0.0882, 0.1314, 0.1554, 0.2286", \
    "0.1494, 0.1578, 0.2124, 0.2508, 0.3528", \
    "0.288, 0.2892, 0.3192, 0.3576, 0.492");
}

cell_rise(delay_template_5x5) {
  index_1 ("0.005, 0.0125, 0.025, 0.075, 0.15");
  index_2 ("0.06, 0.18, 0.42, 0.6, 1.2");
  values ( \
    "0.037639, 0.056898, 0.083401, 0.104927, 0.156652", \
    "0.05258, 0.083003, 0.119028, 0.141927, 0.207952", \
    "0.07402, 0.112622, 0.162437, 0.191122, 0.271755", \
    "0.15767, 0.201007, 0.284096, 0.331746, 0.452958", \
    "0.285016, 0.326868, 0.415086, 0.481337, 0.653064");
}

rise_transition(delay_template_5x5) {
  index_1 ("0.005, 0.0125, 0.025, 0.075, 0.15");
  index_2 ("0.06, 0.18, 0.42, 0.6, 1.2");
  values ( \
    "0.031447, 0.059488, 0.0846, 0.0918, 0.138", \
    "0.047167, 0.0786, 0.1044, 0.1224, 0.1734", \
    "0.072, 0.096, 0.1398, 0.1578, 0.222", \
    "0.1866, 0.1914, 0.2358, 0.2748, 0.3696", \
    "0.3648, 0.3648, 0.384, 0.4146, 0.5388");
}

internal_power() {
  related_pin : "A";
  fall_power(energy_template_5x5) {
    index_1 ("0.005, 0.0125, 0.025, 0.075, 0.15");
    index_2 ("0.06, 0.18, 0.42, 0.6, 1.2");
    values ( \
      "0.009213, 0.004772, 0.00823, 0.018532, 0.054083", \
      "0.009047, 0.005677, 0.005713, 0.015244, 0.049453", \
      "0.008669, 0.006332, 0.002998, 0.01159, 0.04368", \
      "0.007879, 0.007243, 0.001451, 0.004701, 0.030385", \
      "0.007605, 0.007297, 0.003652, 0.000737, 0.020842");
  }

  rise_power(energy_template_5x5) {
    index_1 ("0.005, 0.0125, 0.025, 0.075, 0.15");
    index_2 ("0.06, 0.18, 0.42, 0.6, 1.2");
    values ( \
      "0.023555, 0.029044, 0.041387, 0.051684, 0.087278", \
      "0.023165, 0.028621, 0.039211, 0.048916, 0.083039", \
      "0.023574, 0.02752, 0.036904, 0.045723, 0.077971", \
      "0.024479, 0.025247, 0.032268, 0.039242, 0.066587", \
      "0.024942, 0.025187, 0.029612, 0.034835, 0.057524");
  }
}
}
```

XVII. EVERY GATE MUST BE SIMULATED TO PROVIDE BEHAVIOR OVER INPUT TRANSITION AND LOAD CAPACITANCE

XVIII. ALL ANALOG BLOCKS MUST HAVE ASSOCIATED LIBERTY FILE TO DESCRIBE BEHAVIOR AND TIMING PATHS IF YOU INTEGRATE ANALOG INTO DIGITAL TOP FLOW

XIX. GATE DELAY

1) Delay definitions:

Parameter	Name	Description
t _{pdr}	max rising propagation delay	input to rising output cross 50 %
t _{pdf}	max falling propagation delay	input to falling output cross 50 %
t _{pd}	propagation delay	t _{pdf} = (t _{pdr} + t _{pdf})/2
t _r	rise time	20 % to 80 %
Parameter	Name	Description
t _f	fall time	80 % to 20 %
t _{cdr}	min rising contamination delay	input to rising output cross 50 %
t _{cdf}	min falling contamination delay	input to falling output cross 50 %
t _{cd}	contamination delay	t _{cd} = (t _{cdr} + t _{cdf})/2

XX. DELAY ESTIMATION

How can we get a reasonably accurate hand calculation model of delay?

$$C \approx 1 \text{ fF}/\mu\text{m}$$

$$R \approx 1 \text{ k}\Omega/\mu\text{m}$$

1) Inverter with inverter load:

$$C \approx 1 \text{ fF}/\mu\text{m}$$

,

$$R \approx 1 \text{ k}\Omega/\mu\text{m}$$

$$t_{pd} = R \times 6C = 6RC$$

$$t_{pd} = 6 \times 1 \times 10^3 \times 1 \times 10^{-15} \text{ s}$$

$$t_{pd} = 6 \times 10^{-12} = 6 \text{ ps}$$

XXI. ELMORE DELAY

$$t_{pd} \approx \sum_{\text{nodes}} R_{\text{nodes-to-source}} C_i$$

$$= R_1 C_1 + (R_1 + R_2) C_2 + \dots + (R_1 + R_2 + \dots + R_N) C_N$$

Good enough for hand calculation

XXII. DELAY COMPONENTS

Parasitic delay (p)

p = 9 or 12 RC

Independent of load capacitance

Effort delay (f)

f = 5h RC

Proportional to load capacitance

Let's use process independent unit

$$d = \frac{d_{real}}{\tau}$$

,

$$\tau = 3RC$$

Parasitic delay

$$\Rightarrow p = 12RC/3RC = 4$$

Effort delay

$$\Rightarrow f = 5hRC/3RC = \frac{5}{3}h$$

Delay

$$\Rightarrow d = f + p = \frac{5}{3}h + 4$$

Logical effort (g) is the ratio of the input capacitance of a gate to the input capacitance of an inverter delivering the same output current

Parasitic delay

$$\Rightarrow p = 4$$

Logic effort

$$\Rightarrow g = \frac{5}{3}$$

Electrical effort

$$\Rightarrow h = 1$$

Effort

$$\Rightarrow f = gh$$

Delay

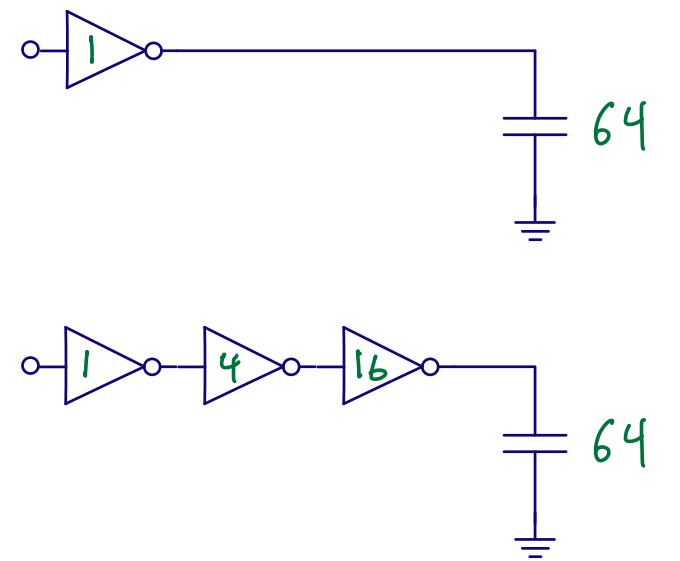
$$\Rightarrow d = f + p = gh + p = 5\frac{2}{3}$$

Real delay

$$\Rightarrow d = 5\frac{2}{3} \times 3 \text{ ps} = 17 \text{ ps}$$

Term	Stage Expression	Path Expression
number of stages	1	N
logical effort	g	$G = \prod (g_i)$
electrical effort	$h = \frac{C_{in}}{C_{out}}$	$H = \frac{C_{out(path)}}{C_{in(path)}}$
branching effort	$b = \frac{C_{onpath} + C_{offpath}}{C_{onpath}}$	$B = \prod b_i$
effort	$f = gh$	$F = GBH$
effort delay	f	$D_F = \sum f_i$
parasitic delay	p	$P = \sum p_i$
delay	$d = f + p$	$D = \sum d_i = D_F + P$

- XXIII. MODERN IC TIMING ANALYSIS REQUIRES
COMPUTERS WITH ADVANCED PROGRAMS³
- XXIV. BEST NUMBER OF STAGES
- XXV. WHICH HAS SHORTEST DELAY?



Term	Stage Expression	Path Expression
number of stages	1	N
logical effort	g	$G = \prod (g_i)$
electrical effort	$h = \frac{C_{in}}{C_{out}}$	$H = \frac{C_{out}(path)}{C_{in}(path)}$
branching effort	$b = \frac{C_{onpath} + C_{offpath}}{C_{onpath}}$	$B = \prod b_i$
effort	$f = gh$	$F = GBH$
effort delay	f	$D_F = \sum f_i$
parsitic delay	p	$P = \sum p_i$
delay	$d = f + p$	$D = \sum d_i = D_F + P$

$$H = C_{cout}/C_{in} = 64$$
$$G = \prod g_i = \prod 1 = 1$$
$$B = 1$$
$$F = GBH = 64$$

One stage

$$f = 64 \Rightarrow D = 64 + 1 = 65$$

³Often called power gating

Three stage with

$$f = 4$$

$$D_F = 12, p = 3 \Rightarrow D = 12 + 3 = 15$$

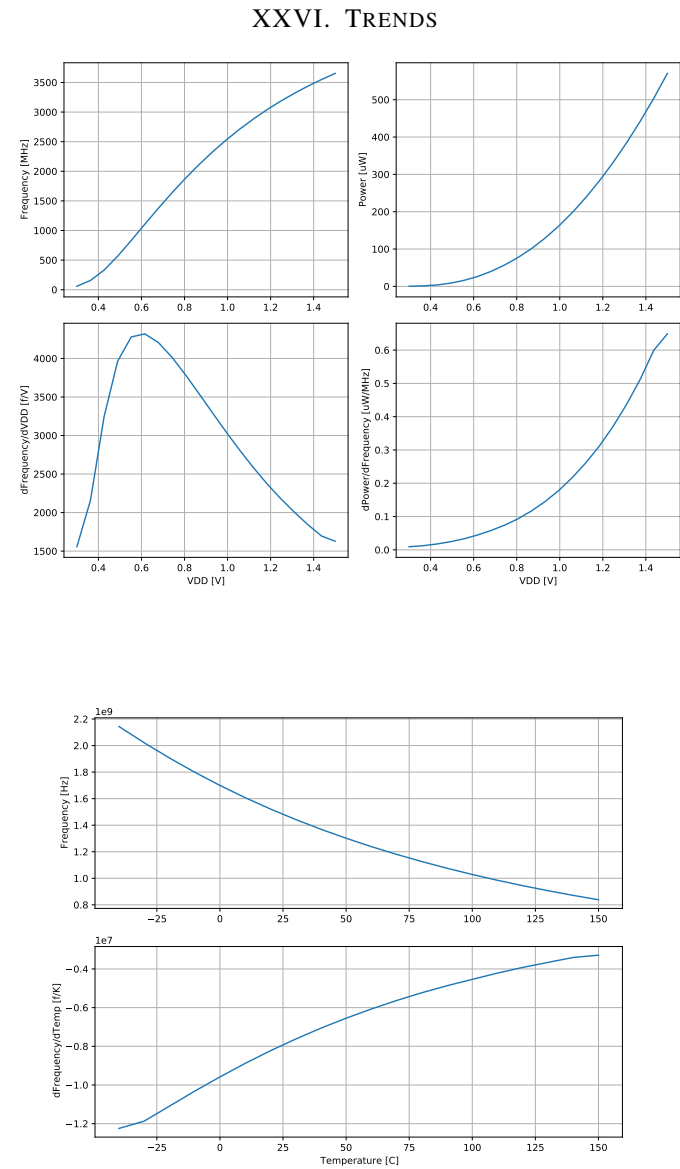
For close to optimal delay, use

$$f = 4$$

(Used to be

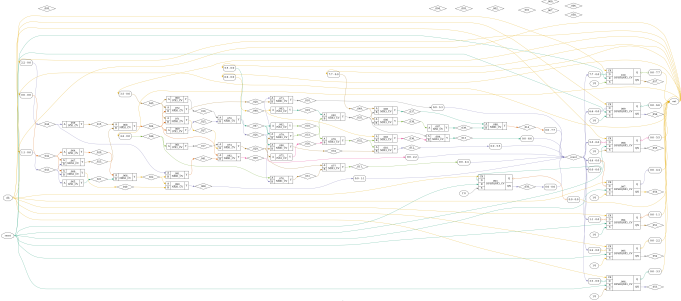
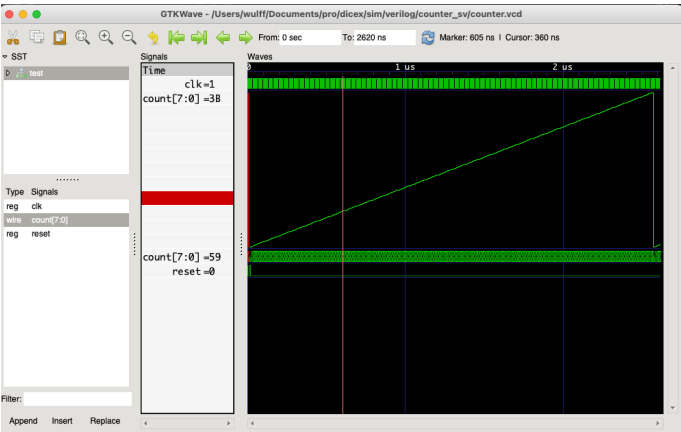
$$f = e$$

)

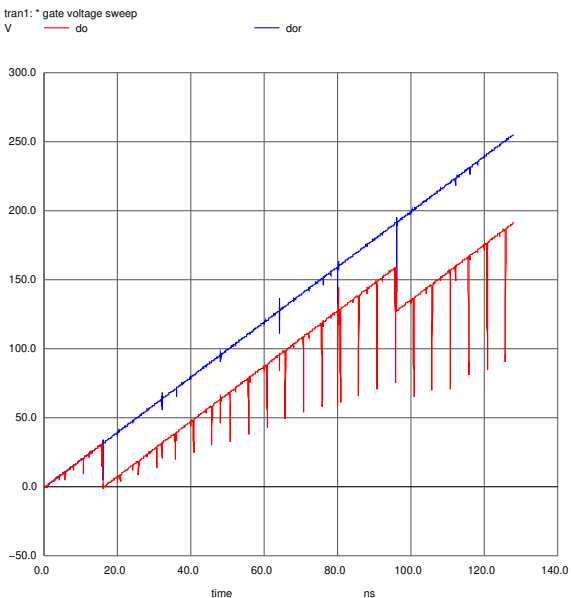
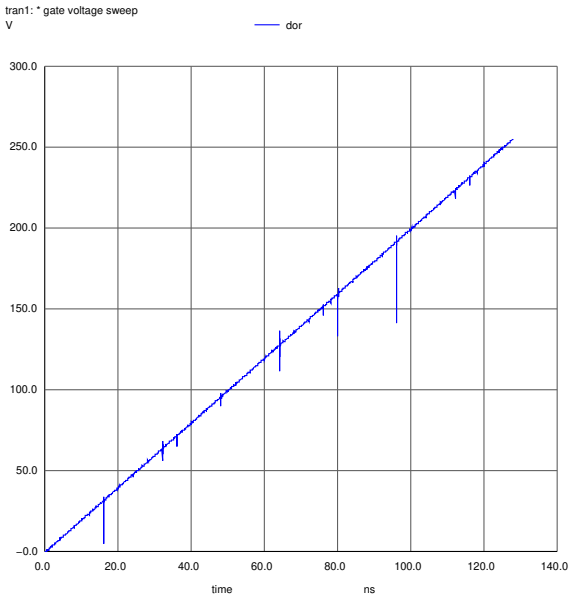


XXVII. ATTACK VECTOR

```
module counter(  
    output logic [WIDTH-1:0] out,  
    input logic clk,  
    input logic reset  
);  
  
parameter WIDTH = 8;  
  
logic [WIDTH-1:0] count;  
always_comb begin  
    count = out + 1;  
end  
  
always_ff @(posedge clk or posedge reset) begin  
    if (reset)  
        out <= 0;  
    else  
        out <= count;  
    end  
end  
  
endmodule // counter
```



```
X14 12 14 15 AVDD AVSS NRX1_CV  
X15 3 13 16 AVDD AVSS NRX1_CV  
X16 16 17 AVDD AVSS IVX1_CV  
X17 out_4 12 18 AVDD AVSS NRX1_CV  
X18 16 18 19 AVDD AVSS NRX1_CV  
X19 4 17 20 AVDD AVSS NRX1_CV  
X20 out_5 16 21 AVDD AVSS NDX1_CV  
X21 out_5 16 22 AVDD AVSS NRX1_CV  
X22 20 22 23 AVDD AVSS NRX1_CV  
X23 5 21 24 AVDD AVSS NRX1_CV  
X24 out_6 20 25 AVDD AVSS NRX1_CV  
X25 24 25 26 AVDD AVSS NRX1_CV  
X26 out_7 24 27 AVDD AVSS NRX1_CV  
X27 out_7 24 28 AVDD AVSS NDX1_CV  
X28 28 29 AVDD AVSS IVX1_CV  
X29 27 29 30 AVDD AVSS NRX1_CV  
X30 out_0 out_1 31 AVDD AVSS NRX1_CV  
X31 8 31 32 AVDD AVSS NRX1_CV  
X32 out_2 8 33 AVDD AVSS NRX1_CV  
X33 10 33 34 AVDD AVSS NRX1_CV  
X34 35 clk AVSS reset out_0 35 AVDD AVSS DFSRQNX1  
X35 32 clk AVSS reset out_1 36 AVDD AVSS DFSRQNX1  
X36 34 clk AVSS reset out_2 37 AVDD AVSS DFSRQNX1  
X37 15 clk AVSS reset out_3 38 AVDD AVSS DFSRQNX1  
X38 19 clk AVSS reset out_4 39 AVDD AVSS DFSRQNX1  
X39 23 clk AVSS reset out_5 40 AVDD AVSS DFSRQNX1  
X40 26 clk AVSS reset out_6 41 AVDD AVSS DFSRQNX1  
X41 30 clk AVSS reset out_7 42 AVDD AVSS DFSRQNX1  
V0 count_0 35 DC 0  
V1 43 out_2 DC 0  
V2 44 out_3 DC 0  
V3 count_3 15 DC 0  
V4 45 out_4 DC 0  
V5 count_4 19 DC 0  
V6 46 out_5 DC 0  
V7 count_5 23 DC 0  
V8 47 out_6 DC 0  
V9 count_6 26 DC 0  
V10 48 out_7 DC 0  
V11 count_7 30 DC 0  
V12 49 out_0 DC 0  
V13 50 out_1 DC 0  
V14 out_2 out_1 32 DC 0  
V15 out_3 out_2 33 DC 0  
V16 out_4 out_3 34 DC 0  
V17 out_5 out_4 35 DC 0  
V18 out_6 out_5 36 DC 0  
V19 out_7 out_6 37 DC 0  
V20 out_0 out_7 38 DC 0  
V21 out_1 out_0 39 DC 0  
V22 out_2 out_1 40 DC 0  
V23 out_3 out_2 41 DC 0  
V24 out_4 out_3 42 DC 0  
V25 out_5 out_4 43 DC 0  
V26 out_6 out_5 44 DC 0  
V27 out_7 out_6 45 DC 0  
V28 out_0 out_7 46 DC 0  
V29 out_1 out_0 47 DC 0  
V30 out_2 out_1 48 DC 0  
V31 out_3 out_2 49 DC 0  
V32 out_4 out_3 50 DC 0  
V33 out_5 out_4 51 DC 0  
V34 out_6 out_5 52 DC 0  
V35 out_7 out_6 53 DC 0  
V36 out_0 out_7 54 DC 0  
V37 out_1 out_0 55 DC 0  
V38 out_2 out_1 56 DC 0  
V39 out_3 out_2 57 DC 0  
V40 out_4 out_3 58 DC 0  
V41 out_5 out_4 59 DC 0  
V42 out_6 out_5 60 DC 0  
V43 out_7 out_6 61 DC 0  
V44 out_0 out_7 62 DC 0  
V45 out_1 out_0 63 DC 0  
V46 out_2 out_1 64 DC 0  
V47 out_3 out_2 65 DC 0  
V48 out_4 out_3 66 DC 0  
V49 out_5 out_4 67 DC 0  
V50 out_6 out_5 68 DC 0  
V51 out_7 out_6 69 DC 0  
V52 out_0 out_7 70 DC 0  
V53 out_1 out_0 71 DC 0  
V54 out_2 out_1 72 DC 0  
V55 out_3 out_2 73 DC 0  
V56 out_4 out_3 74 DC 0  
V57 out_5 out_4 75 DC 0  
V58 out_6 out_5 76 DC 0  
V59 out_7 out_6 77 DC 0  
V60 out_0 out_7 78 DC 0  
V61 out_1 out_0 79 DC 0  
V62 out_2 out_1 80 DC 0  
V63 out_3 out_2 81 DC 0  
V64 out_4 out_3 82 DC 0  
V65 out_5 out_4 83 DC 0  
V66 out_6 out_5 84 DC 0  
V67 out_7 out_6 85 DC 0  
V68 out_0 out_7 86 DC 0  
V69 out_1 out_0 87 DC 0  
V70 out_2 out_1 88 DC 0  
V71 out_3 out_2 89 DC 0  
V72 out_4 out_3 90 DC 0  
V73 out_5 out_4 91 DC 0  
V74 out_6 out_5 92 DC 0  
V75 out_7 out_6 93 DC 0  
V76 out_0 out_7 94 DC 0  
V77 out_1 out_0 95 DC 0  
V78 out_2 out_1 96 DC 0  
V79 out_3 out_2 97 DC 0  
V80 out_4 out_3 98 DC 0  
V81 out_5 out_4 99 DC 0  
V82 out_6 out_5 100 DC 0  
V83 out_7 out_6 101 DC 0  
V84 out_0 out_7 102 DC 0  
V85 out_1 out_0 103 DC 0  
V86 out_2 out_1 104 DC 0  
V87 out_3 out_2 105 DC 0  
V88 out_4 out_3 106 DC 0  
V89 out_5 out_4 107 DC 0  
V90 out_6 out_5 108 DC 0  
V91 out_7 out_6 109 DC 0  
V92 out_0 out_7 110 DC 0  
V93 out_1 out_0 111 DC 0  
V94 out_2 out_1 112 DC 0  
V95 out_3 out_2 113 DC 0  
V96 out_4 out_3 114 DC 0  
V97 out_5 out_4 115 DC 0  
V98 out_6 out_5 116 DC 0  
V99 out_7 out_6 117 DC 0  
V100 out_0 out_7 118 DC 0  
V101 out_1 out_0 119 DC 0  
V102 out_2 out_1 120 DC 0  
V103 out_3 out_2 121 DC 0  
V104 out_4 out_3 122 DC 0  
V105 out_5 out_4 123 DC 0  
V106 out_6 out_5 124 DC 0  
V107 out_7 out_6 125 DC 0  
V108 out_0 out_7 126 DC 0  
V109 out_1 out_0 127 DC 0  
V110 out_2 out_1 128 DC 0  
V111 out_3 out_2 129 DC 0  
V112 out_4 out_3 130 DC 0  
V113 out_5 out_4 131 DC 0  
V114 out_6 out_5 132 DC 0  
V115 out_7 out_6 133 DC 0  
V116 out_0 out_7 134 DC 0  
V117 out_1 out_0 135 DC 0  
V118 out_2 out_1 136 DC 0  
V119 out_3 out_2 137 DC 0  
V120 out_4 out_3 138 DC 0  
V121 out_5 out_4 139 DC 0  
V122 out_6 out_5 140 DC 0  
V123 out_7 out_6 141 DC 0  
V124 out_0 out_7 142 DC 0  
V125 out_1 out_0 143 DC 0  
V126 out_2 out_1 144 DC 0  
V127 out_3 out_2 145 DC 0  
V128 out_4 out_3 146 DC 0  
V129 out_5 out_4 147 DC 0  
V130 out_6 out_5 148 DC 0  
V131 out_7 out_6 149 DC 0  
V132 out_0 out_7 150 DC 0  
V133 out_1 out_0 151 DC 0  
V134 out_2 out_1 152 DC 0  
V135 out_3 out_2 153 DC 0  
V136 out_4 out_3 154 DC 0  
V137 out_5 out_4 155 DC 0  
V138 out_6 out_5 156 DC 0  
V139 out_7 out_6 157 DC 0  
V140 out_0 out_7 158 DC 0  
V141 out_1 out_0 159 DC 0  
V142 out_2 out_1 160 DC 0  
V143 out_3 out_2 161 DC 0  
V144 out_4 out_3 162 DC 0  
V145 out_5 out_4 163 DC 0  
V146 out_6 out_5 164 DC 0  
V147 out_7 out_6 165 DC 0  
V148 out_0 out_7 166 DC 0  
V149 out_1 out_0 167 DC 0  
V150 out_2 out_1 168 DC 0  
V151 out_3 out_2 169 DC 0  
V152 out_4 out_3 170 DC 0  
V153 out_5 out_4 171 DC 0  
V154 out_6 out_5 172 DC 0  
V155 out_7 out_6 173 DC 0  
V156 out_0 out_7 174 DC 0  
V157 out_1 out_0 175 DC 0  
V158 out_2 out_1 176 DC 0  
V159 out_3 out_2 177 DC 0  
V160 out_4 out_3 178 DC 0  
V161 out_5 out_4 179 DC 0  
V162 out_6 out_5 180 DC 0  
V163 out_7 out_6 181 DC 0  
V164 out_0 out_7 182 DC 0  
V165 out_1 out_0 183 DC 0  
V166 out_2 out_1 184 DC 0  
V167 out_3 out_2 185 DC 0  
V168 out_4 out_3 186 DC 0  
V169 out_5 out_4 187 DC 0  
V170 out_6 out_5 188 DC 0  
V171 out_7 out_6 189 DC 0  
V172 out_0 out_7 190 DC 0  
V173 out_1 out_0 191 DC 0  
V174 out_2 out_1 192 DC 0  
V175 out_3 out_2 193 DC 0  
V176 out_4 out_3 194 DC 0  
V177 out_5 out_4 195 DC 0  
V178 out_6 out_5 196 DC 0  
V179 out_7 out_6 197 DC 0  
V180 out_0 out_7 198 DC 0  
V181 out_1 out_0 199 DC 0  
V182 out_2 out_1 200 DC 0  
V183 out_3 out_2 201 DC 0  
V184 out_4 out_3 202 DC 0  
V185 out_5 out_4 203 DC 0  
V186 out_6 out_5 204 DC 0  
V187 out_7 out_6 205 DC 0  
V188 out_0 out_7 206 DC 0  
V189 out_1 out_0 207 DC 0  
V190 out_2 out_1 208 DC 0  
V191 out_3 out_2 209 DC 0  
V192 out_4 out_3 210 DC 0  
V193 out_5 out_4 211 DC 0  
V194 out_6 out_5 212 DC 0  
V195 out_7 out_6 213 DC 0  
V196 out_0 out_7 214 DC 0  
V197 out_1 out_0 215 DC 0  
V198 out_2 out_1 216 DC 0  
V199 out_3 out_2 217 DC 0  
V200 out_4 out_3 218 DC 0  
V201 out_5 out_4 219 DC 0  
V202 out_6 out_5 220 DC 0  
V203 out_7 out_6 221 DC 0  
V204 out_0 out_7 222 DC 0  
V205 out_1 out_0 223 DC 0  
V206 out_2 out_1 224 DC 0  
V207 out_3 out_2 225 DC 0  
V208 out_4 out_3 226 DC 0  
V209 out_5 out_4 227 DC 0  
V210 out_6 out_5 228 DC 0  
V211 out_7 out_6 229 DC 0  
V212 out_0 out_7 230 DC 0  
V213 out_1 out_0 231 DC 0  
V214 out_2 out_1 232 DC 0  
V215 out_3 out_2 233 DC 0  
V216 out_4 out_3 234 DC 0  
V217 out_5 out_4 235 DC 0  
V218 out_6 out_5 236 DC 0  
V219 out_7 out_6 237 DC 0  
V220 out_0 out_7 238 DC 0  
V221 out_1 out_0 239 DC 0  
V222 out_2 out_1 240 DC 0  
V223 out_3 out_2 241 DC 0  
V224 out_4 out_3 242 DC 0  
V225 out_5 out_4 243 DC 0  
V226 out_6 out_5 244 DC 0  
V227 out_7 out_6 245 DC 0  
V228 out_0 out_7 246 DC 0  
V229 out_1 out_0 247 DC 0  
V230 out_2 out_1 248 DC 0  
V231 out_3 out_2 249 DC 0  
V232 out_4 out_3 250 DC 0  
V233 out_5 out_4 251 DC 0  
V234 out_6 out_5 252 DC 0  
V235 out_7 out_6 253 DC 0  
V236 out_0 out_7 254 DC 0  
V237 out_1 out_0 255 DC 0  
V238 out_2 out_1 256 DC 0  
V239 out_3 out_2 257 DC 0  
V240 out_4 out_3 258 DC 0  
V241 out_5 out_4 259 DC 0  
V242 out_6 out_5 260 DC 0  
V243 out_7 out_6 261 DC 0  
V244 out_0 out_7 262 DC 0  
V245 out_1 out_0 263 DC 0  
V246 out_2 out_1 264 DC 0  
V247 out_3 out_2 265 DC 0  
V248 out_4 out_3 266 DC 0  
V249 out_5 out_4 267 DC 0  
V250 out_6 out_5 268 DC 0  
V251 out_7 out_6 269 DC 0  
V252 out_0 out_7 270 DC 0  
V253 out_1 out_0 271 DC 0  
V254 out_2 out_1 272 DC 0  
V255 out_3 out_2 273 DC 0  
V256 out_4 out_3 274 DC 0  
V257 out_5 out_4 275 DC 0  
V258 out_6 out_5 276 DC 0  
V259 out_7 out_6 277 DC 0  
V260 out_0 out_7 278 DC 0  
V261 out_1 out_0 279 DC 0  
V262 out_2 out_1 280 DC 0  
V263 out_3 out_2 281 DC 0  
V264 out_4 out_3 282 DC 0  
V265 out_5 out_4 283 DC 0  
V266 out_6 out_5 284 DC 0  
V267 out_7 out_6 285 DC 0  
V268 out_0 out_7 286 DC 0  
V269 out_1 out_0 287 DC 0  
V270 out_2 out_1 288 DC 0  
V271 out_3 out_2 289 DC 0  
V272 out_4 out_3 290 DC 0  
V273 out_5 out_4 291 DC 0  
V274 out_6 out_5 292 DC 0  
V275 out_7 out_6 293 DC 0  
V276 out_0 out_7 294 DC 0  
V277 out_1 out_0 295 DC 0  
V278 out_2 out_1 296 DC 0  
V279 out_3 out_2 297 DC 0  
V280 out_4 out_3 298 DC 0  
V281 out_5 out_4 299 DC 0  
V282 out_6 out_5 300 DC 0  
V283 out_7 out_6 301 DC 0  
V284 out_0 out_7 302 DC 0  
V285 out_1 out_0 303 DC 0  
V286 out_2 out_1 304 DC 0  
V287 out_3 out_2 305 DC 0  
V288 out_4 out_3 306 DC 0  
V289 out_5 out_4 307 DC 0  
V290 out_6 out_5 308 DC 0  
V291 out_7 out_6 309 DC 0  
V292 out_0 out_7 310 DC 0  
V293 out_1 out_0 311 DC 0  
V294 out_2 out_1 312 DC 0  
V295 out_3 out_2 313 DC 0  
V296 out_4 out_3 314 DC 0  
V297 out_5 out_4 315 DC 0  
V298 out_6 out_5 316 DC 0  
V299 out_7 out_6 317 DC 0  
V300 out_0 out_7 318 DC 0  
V301 out_1 out_0 319 DC 0  
V302 out_2 out_1 320 DC 0  
V303 out_3 out_2 321 DC 0  
V304 out_4 out_3 322 DC 0  
V305 out_5 out_4 323 DC 0  
V306 out_6 out_5 324 DC 0  
V307 out_7 out_6 325 DC 0  
V308 out_0 out_7 326 DC 0  
V309 out_1 out_0 327 DC 0  
V310 out_2 out_1 328 DC 0  
V311 out_3 out_2 329 DC 0  
V312 out_4 out_3 330 DC 0  
V313 out_5 out_4 331 DC 0  
V314 out_6 out_5 332 DC 0  
V315 out_7 out_6 333 DC 0  
V316 out_0 out_7 334 DC 0  
V317 out_1 out_0 335 DC 0  
V318 out_2 out_1 336 DC 0  
V319 out_3 out_2 337 DC 0  
V320 out_4 out_3 338 DC 0  
V321 out_5 out_4 339 DC 0  
V322 out_6 out_5 340 DC 0  
V323 out_7 out_6 341 DC 0  
V324 out_0 out_7 342 DC 0  
V325 out_1 out_0 343 DC 0  
V326 out_2 out_1 344 DC 0  
V327 out_3 out_2 345 DC 0  
V328 out_4 out_3 346 DC 0  
V329 out_5 out_4 347 DC 0  
V330 out_6 out_5 348 DC 0  
V331 out_7 out_6 349 DC 0  
V332 out_0 out_7 350 DC 0  
V333 out_1 out_0 351 DC 0  
V334 out_2 out_1 352 DC 0  
V335 out_3 out_2 353 DC 0  
V336 out_4 out_3 354 DC 0  
V337 out_5 out_4 355 DC 0  
V338 out_6 out_5 356 DC 0  
V339 out_7 out_6 357 DC 0  
V340 out_0 out_7 358 DC 0  
V341 out_1 out_0 359 DC 0  
V342 out_2 out_1 360 DC 0  
V343 out_3 out_2 361 DC 0  
V344 out_4 out_3 362 DC 0  
V345 out_5 out_4 363 DC 0  
V346 out_6 out_5 364 DC 0  
V347 out_7 out_6 365 DC 0  
V348 out_0 out_7 366 DC 0  
V349 out_1 out_0 367 DC 0  
V350 out_2 out_1 368 DC 0  
V351 out_3 out_2 369 DC 0  
V352 out_4 out_3 370 DC 0  
V353 out_5 out_4 371 DC 0  
V354 out_6 out_5 372 DC 0  
V355 out_7 out_6 373 DC 0  
V356 out_0 out_7 374 DC 0  
V357 out_1 out_0 375 DC 0  
V358 out_2 out_1 376 DC 0  
V359 out_3 out_2 377 DC 0  
V360 out_4 out_3 378 DC 0  
V361 out_5 out_4 379 DC 0  
V362 out_6 out_5 380 DC 0  
V363 out_7 out_6 381 DC 0  
V364 out_0 out_7 382 DC 0  
V365 out_1 out_0 383 DC 0  
V366 out_2 out_1 384 DC 0  
V367 out_3 out_2 385 DC 0  
V368 out_4 out_3 386 DC 0  
V369 out_5 out_4 387 DC 0  
V370 out_6 out_5 388 DC 0  
V371 out_7 out_6 389 DC 0  
V372 out_0 out_7 390 DC 0  
V373 out_1 out_0 391 DC 0  
V374 out_2 out_1 392 DC 0  
V375 out_3 out_2 393 DC 0  
V376 out_4 out_3 394 DC 0  
V377 out_5 out_4 395 DC 0  
V378 out_6 out_5 396 DC 0  
V379 out_7 out_6 397 DC 0  
V380 out_0 out_7 398 DC 0  
V381 out_1 out_0 399 DC 0  
V382 out_2 out_1 400 DC 0  
V383 out_3 out_2 401 DC 0  
V384 out_4 out_3 402 DC 0  
V385 out_5 out_4 403 DC 0  
V386 out_6 out_5 404 DC 0  
V387 out_7 out_6 405 DC 0  
V388 out_0 out_7 406 DC 0  
V389 out_1 out_0 407 DC 0  
V390 out_2 out_1 408 DC 0  
V391 out_3 out_2 409 DC 0  
V392 out_4 out_3 410 DC 0  
V393 out_5 out_4 411 DC 0  
V394 out_6 out_5 412 DC 0  
V395 out_7 out_6 413 DC 0  
V396 out_0 out_7 414 DC 0  
V397 out_1 out_0 415 DC 0  
V398 out_2 out_1 416 DC 0  
V399 out_3 out_2 417 DC 0  
V400 out_4 out_3 418 DC 0  
V401 out_5 out_4 419 DC 0  
V402 out_6 out_5 420 DC 0  
V403 out_7 out_6 421 DC 0  
V404 out_0 out_7 422 DC 0  
V405 out_1 out_0 423 DC 0  
V406 out_2 out_1 424 DC 0  
V407 out_3 out_2 425 DC 0  
V408 out_4 out_3 426 DC 0  
V409 out_5 out_4 427 DC 0  
V410 out_6 out_5 428 DC 0  
V411 out_7 out_6 429 DC 0  
V412 out_0 out_7 430 DC 0  
V413 out_1 out_0 431 DC 0  
V414 out_2 out_1 432 DC 0  
V415 out_3 out_2 433 DC 0  
V416 out_4 out_3 434 DC 0  
V417 out_5 out_4 435 DC 0  
V418 out_6 out_5 436 DC 0  
V419 out_7 out_6 437 DC 0  
V420 out_0 out_7 438 DC 0  
V421 out_1 out_0 439 DC 0  
V422 out_2 out_1 440 DC 0  
V423 out_3 out_2 441 DC 0  
V424 out_4 out_3 442 DC 0  
V425 out_5 out_4 443 DC 0  
V426 out_6 out_5 444 DC 0  
V427 out_7 out_6 445 DC 0  
V428 out_0 out_7 446 DC 0  
V429 out_1 out_0 447 DC 0  
V430 out_2 out_1 448 DC 0  
V431 out_3 out_2 449 DC 0  
V432 out_4 out_3 450 DC 0  
V433 out_5 out_4 451 DC 0  
V434 out_6 out_5 452 DC 0  
V435 out_7 out_6 453 DC 0  
V436 out_0 out_7 454 DC 0  
V437 out_1 out_0 455 DC 0  
V438 out_2 out_1 456 DC 0  
V439 out_3 out_2 457 DC 0  
V440 out_4 out_3 458 DC 0  
V441 out_5 out_4 459 DC 0  
V442 out_6 out_5 460 DC 0  
V443 out_7 out_6 461 DC 0  
V444 out_0 out_7 462 DC 0  
V445 out_1 out_0 463 DC 0  
V446 out_2 out_1 464 DC 0  
V447 out_3 out_2 465 DC 0  
V448 out_4 out_3 466 DC 0  
V449 out_5 out_4 467 DC 0  
V450 out_6 out_5 468 DC 0  
V451 out_7 out_6 469 DC 0  
V452 out_0 out_7 470 DC 0  
V453 out_1 out_0 471 DC 0  
V454 out_2 out_1 472 DC 0  
V455 out_3 out_2 473 DC 0  
V456 out_4 out_3 474 DC 0  
V457 out_5 out_4 475 DC 0  
V458 out_6 out_5 476 DC 0  
V459 out_7 out_6 477 DC 0  
V460 out_0 out_7 478 DC 0  
V461 out_1 out_0 479 DC 0  
V462 out_2 out_1 480 DC 0  
V463 out_3 out_2 481 DC 0  
V464 out_4 out_3 482 DC 0  
V465 out_5 out_4 483 DC 0  
V466 out_6 out_5 484 DC 0  
V467 out_7 out_6 485 DC 0  
V468 out_0 out_7 486 DC 0  
V469 out_1 out_0 487 DC 0  
V470 out_2 out_1 488 DC 0  
V471 out_3 out_2 489 DC 0  
V472 out_4 out_3 490 DC 0  
V473 out_5 out_4 491 DC 0  
V474 out_6 out_5 492 DC 0  
V475 out_7 out_6 493 DC 0  
V476 out_0 out_7 494 DC 0  
V477 out_1 out_0 495 DC 0  
V478 out_2 out_1 496 DC 0  
V479 out_3 out_2 497 DC 0  
V480 out_4 out_3 498 DC 0  
V481 out_5 out_4 499 DC 0  
V482 out_6 out_5 500 DC 0  
V483 out_7 out_6 501 DC 0  
V484 out_0 out_7 502 DC 0  
V485 out_1 out_0 503 DC 0  
V486 out_2 out_1 504 DC 0  
V487 out_3 out_2 505 DC 0  
V488 out_4 out_3 506 DC 0  
V489 out_5 out_4 507 DC 0  
V490 out_6 out_5 508 DC 0  
V491 out_7 out_6 509 DC 0  
V492 out_0 out_7 510 DC 0  
V493 out_1 out_0 511 DC 0  
V494 out_2 out_1 512 DC 0  
V495 out_3 out_2 513 DC 0  
V496 out_4 out_3 514 DC 0  
V497 out_5 out_4 515 DC 0  
V498 out_6 out_5 516 DC 0  
V499 out_7 out_6 517 DC 0  
V500 out_0 out_7 518 DC 0  
V501 out_1 out_0 519 DC 0  
V502 out_2 out_1 520 DC 0  
V503 out_3 out_2 521 DC 0  
V504 out_4 out_3 522 DC 0  
V505 out_5 out_4 523 DC 0  
V506 out_6 out_5 524 DC 0  
V507 out_7 out_6 525 DC 0  
V508 out_0 out_7 526 DC 0  
V509 out_1 out_0 527 DC 0  
V510 out_2 out_1 528 DC 0  
V511 out_3 out_2 529 DC 0  
V512 out_4 out_3 530 DC 0  
V513 out_5 out_4 531 DC 0  
V514 out_6 out_5 532 DC 0  
V515 out_7 out_6 533 DC 0  
V516 out_0 out_7 534 DC 0  
V517 out_1 out_0 535 DC 0  
V518 out_2 out_1 536 DC 0  
V519 out_3 out_2 537 DC 0  
V520 out_4 out_3 538 DC 0  
V521 out_5 out_4 539 DC 0  
V522 out_6 out_5 540 DC 0  
V523 out_7 out_6 541 DC 0  
V524 out_0 out_7 542 DC 0  
V525 out_1 out_0 543 DC 0  
V526 out_2 out_1 544 DC 0  
V527 out_3 out_2 545 DC 0  
V528 out_4 out_3 546 DC 0  
V529 out_5 out_4 547 DC 0  
V530 out_6 out_5 548 DC 0  
V531 out_7 out_6 549 DC 0  
V532 out_0 out_7 550 DC 0  
V533 out_1 out_0 551 DC 0  
V534 out_2 out_1 552 DC 0  
V535 out_3 out_2 553 DC 0  
V536 out_4 out_3 554 DC 0  
V537 out_5 out_4 555 DC 0  
V538 out_6 out_5 556 DC 0  
V539 out_7 out_6 557 DC 0  
V540 out_0 out_7 558 DC 0  
V541 out_1 out_0 559 DC 0  
V542 out_2 out_1 560 DC 0  
V543 out_3 out_2 561 DC 0  
V544 out_4 out_3 562 DC 0  
V545 out_5 out_4 563 DC 0  
V546 out_6 out_5 564 DC 0  
V547 out_7 out_6 565 DC 0  
V548 out_0 out_7 566 DC 0  
V549 out_1 out_0 567 DC 0  
V550 out_2 out_1 568 DC 0  
V551 out_3 out_2 569 DC 0  
V552 out_4 out_3 570 DC 0  
V553 out_5 out_4 571 DC 0  
V554 out_6 out_5 572 DC 0  
V555 out_7 out_6 573 DC 0  
V556 out_0 out_7 574 DC 0  
V557 out_1 out_0 575 DC 0  
V558 out_2 out_1 576 DC 0  
V559 out_3 out_2 577 DC 0  
V560 out_4 out_3 578 DC 0  
V561 out_5 out_4 579 DC 0  
V562 out_6 out_5 580 DC 0  
V563 out_7 out_6 581 DC 0  
V564 out_0 out_7 582 DC 0  
V565 out_1 out_0 583 DC 0  
V566 out_2 out_1 584 DC 0  
V567 out_3 out_2 585 DC 0  
V568 out_4 out_3 586 DC 0  
V569 out_5 out_4 587 DC 0  
V570 out_6 out_5 588 DC 0  
V571 out_7 out_6 589 DC 0  
V572 out_0 out_7 590 DC 0  
V573 out_1 out_0 591 DC 0  
V574 out_2 out_1 592 DC 0  
V575 out_3 out_2 593 DC 0  
V576 out_4 out_3 594 DC 0  
V577 out_5 out_4 595 DC 0  
V578 out_6 out_5 596 DC 0  
V579 out_7 out_6 597 DC 0  
V580 out_0 out_7 598 DC 0  
V581 out_1 out_0 599 DC 0  
V582 out_2 out_1 600 DC 0  
V583 out_3 out_2 601 DC 0  
V584 out_4 out_3 602 DC 0  
V585 out_5 out_4 603 DC 0  
V586 out_6 out_5 604 DC 0  
V587 out_7 out_6 605 DC 0  
V588 out_0 out_7 606 DC 0  
V589 out_1 out_0 607 DC 0  
V590 out_2 out_1 608 DC 0  
V591 out_3 out_2 609 DC 0  
V592 out_4 out_3 610 DC 0  
V593 out_5 out_4 611 DC 0  
V594 out_6 out_5 612 DC 0  
V595 out_7 out_6 613 DC 0  
V596 out_0 out_7 614 DC 0  
V597 out_1 out_0 615 DC 0  
V598 out_2 out_1 616 DC 0  
V599 out_3 out_2
```



dicex/sim/verilog/counter_sv/counter_attack_tb.cir

VDDA AVDD_ATTACK 0 dc 0.5 pulse(1.5 0.6 tcd trf trf tapw taper)

ChipWhisperer

The ChipWhisperer® ecosystem presents the first open-source, low-cost solution to expose weaknesses that exist in embedded systems all around us.

Software Documentation Hardware Documentation

Single Board Solutions

Starter Kits

Target Boards

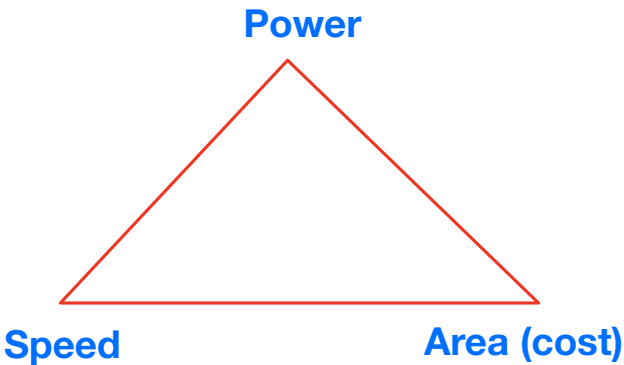
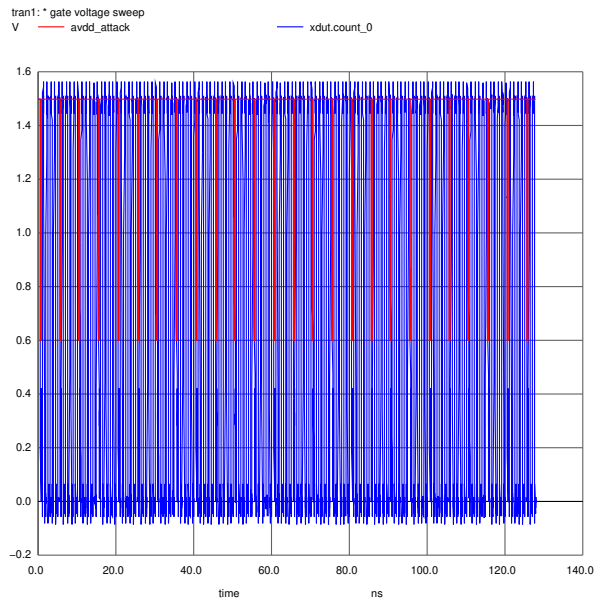
CW308 UFO Targets


conduct side-channel
power analysis and fault
injection attacks


uncover vulnerabilities
in your own
embedded systems


implement
robust security
countermeasure

XXVIII. PICK TWO



XXIX. POWER

XXX. WHAT IS POWER?

Instantaneous power:

$$P(t) = I(t)V(t)$$

Energy :

$$\int_0^T P(t)dt$$

[J]

Average power:

$$\frac{1}{T} \int_0^T P(t)dt$$

[W or J/s]

XXXI. POWER DISSIPATED IN A RESISTOR

Ohm's Law

$$V_R = I_R R$$

$$P_R = V_R I_R = I_R^2 R = \frac{V_R^2}{R}$$

XXXII. CHARGING A CAPACITOR TO VDD

Capacitor differential equation

$$I_C = C \frac{dV}{dt}$$

$$E_C = \int_0^\infty I_C V_C dt = \int_0^\infty C \frac{dV}{dt} V_C dt = \int_0^{V_C} C V dV = C \left[\frac{V^2}{2} \right]_0^{V_{DD}}$$

$$E_C = \frac{1}{2} C V_{DD}^2$$

XXXIII. ENERGY TO CHARGE A CAPACITOR TO A VOLTAGE VDD

$$E_C = \frac{1}{2} C V_{DD}^2$$

$$I_{VDD} = I_C = C \frac{dV}{dt}$$

$$E_{VDD} = \int_0^\infty I_{VDD} V_{DD} dt = \int_0^\infty C \frac{dV}{dt} V_{DD} dt = C V_{DD} \int_0^{V_{DD}} dV = C V_{DD}^2$$

Only half the energy is stored on the capacitor, the rest is dissipated in the PMOS

XXXIV. DISCHARGING A CAPACITOR TO 0

$$E_C = \frac{1}{2} C V_{DD}^2$$

Voltage is pulled to ground, and the power is dissipated in the NMOS

XXXV. POWER CONSUMPTION OF DIGITAL CIRCUITS

$$E_{VDD} = C V_{DD}^2 f$$

In a clock distribution network (chain of inverters), every output is charged once per clock cycle

$$P_{VDD} = C V_{DD}^2 f$$

XXXVI. SOURCES OF POWER DISSIPATION IN CMOS LOGIC

$$P_{total} = P_{dynamic} + P_{static}$$

Dynamic power dissipation

Charging and discharging load capacitances

short-circuit current, when PMOS and NMOS conduct at the same time

$$P_{dynamic} = P_{switching} + P_{shortcircuit}$$

Static power dissipation

Subthreshold leakage in OFF transistors

Gate leakage (tunneling current) through gate dielectric

Source/drain reverse bias PN junction leakage

$$P_{static} = (I_{sub} + I_{gate} + I_{pn}) V_{DD}$$

XXXVII. SWITCHING POWER IN LOGIC GATES

Only output node transitions from low to high consume power from

$$V_{DD}$$

Define

$$P_i$$

to be the probability that a node is 1

Define $\overline{P}_i = 1 - P_i$

$$\overline{P}_i = 1 - P_i$$

to be the probability that a node is 0

Define **activity factor** (

$$\alpha_i$$

) as the **probability of switching a node from 0 to 1**

If the probability is uncorrelated from cycle to cycle

$$\alpha_i = \overline{P}_i P_i$$

XXXVIII. SWITCHING PROBABILITY

Random data

$P = 0.5$

,

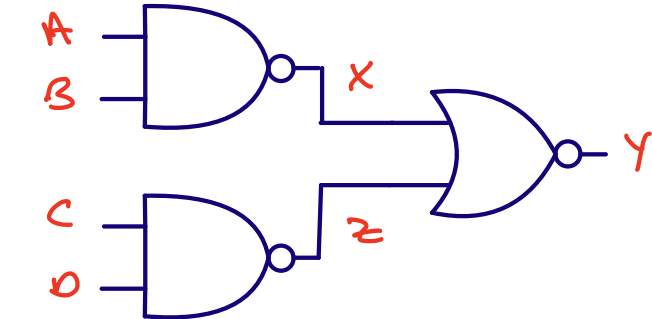
$\alpha = 0.25$

Clocks

$\alpha = 1$

Gate	P_Y
AND2	$P_A P_B$
OR2	$1 - \bar{P}_A \bar{P}_B$
NAND2	$1 - P_A P_B$
NOR2	$\bar{P}_A \bar{P}_B$
XOR2	$P_A \bar{P}_B + \bar{P}_A P_B$

Gate	P_Y
AND2	$P_A P_B$
OR2	$1 - \bar{P}_A \bar{P}_B$
NAND2	$1 - P_A P_B$
NOR2	$\bar{P}_A \bar{P}_B$
XOR2	$P_A \bar{P}_B + \bar{P}_A P_B$



Assume

$P = P_A = P_B = P_C = P_D = \frac{1}{2}$

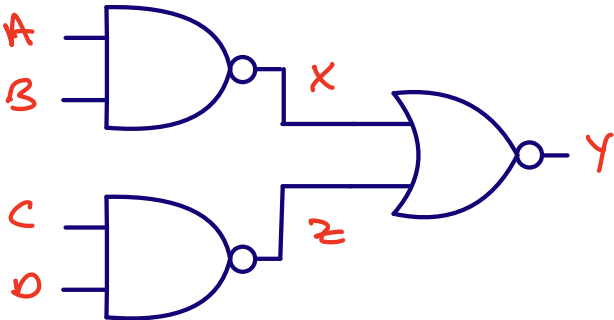
$P_X = P_Z = 1 - PP = 1 - \frac{1}{4} = \frac{3}{4}$

$\overline{P_X} = \overline{P_Y} = \frac{1}{4}$

$P_Y = \frac{1}{4} \times \frac{1}{4} = \frac{1}{16}$

$\alpha = \frac{1}{16} \left(1 - \frac{1}{16} \right) = \frac{15}{16} \frac{1}{16} = \frac{15}{256}$

Gate	P_Y
AND2	$P_A P_B$
OR2	$1 - \bar{P}_A \bar{P}_B$
NAND2	$1 - P_A P_B$
NOR2	$\bar{P}_A \bar{P}_B$
XOR2	$P_A \bar{P}_B + \bar{P}_A P_B$



$\overline{\overline{AB} + \overline{CD}}$

Use De Morgan first

$\overline{A + B} = \bar{A} \cdot \bar{B}$

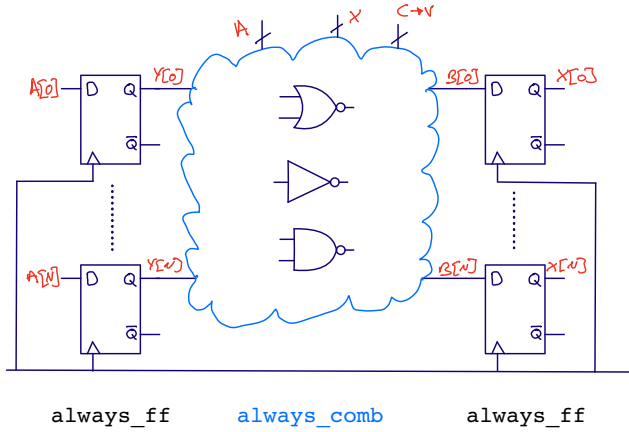
$\overline{\overline{AB} + \overline{CD}} = \overline{\overline{ABCD}} = ABCD$

$\Rightarrow P_Y = P_A P_B P_C P_D = \left(\frac{1}{2} \right)^4 = \frac{1}{16}$

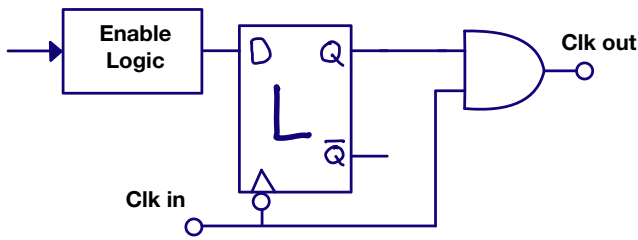
$P_{tot} = \alpha C V_{DD}^2 f$

XXXIX. STRATEGIES TO REDUCE DYNAMIC POWER

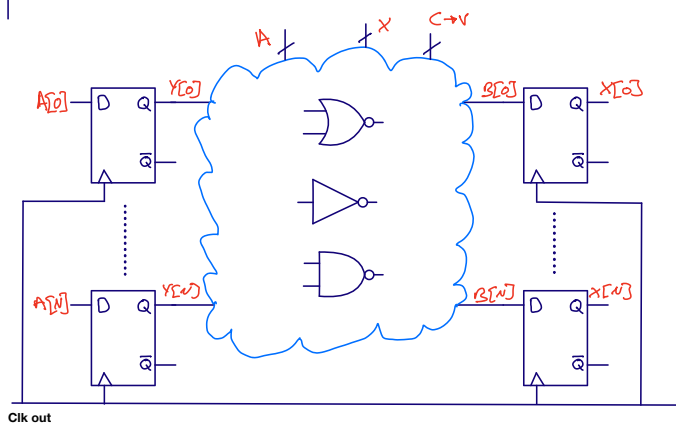
1. Stop clock
2. Stop activity
3. Reduce clock frequency
4. Turn off VDD
5. Reduce VDD



A. Stop clock ⁴

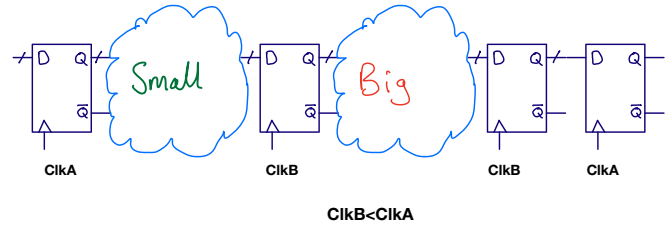


B. Stop activity

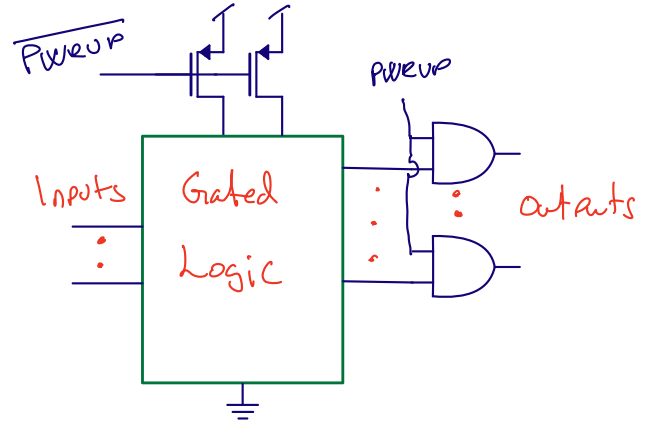


⁴Often called *clock gating*

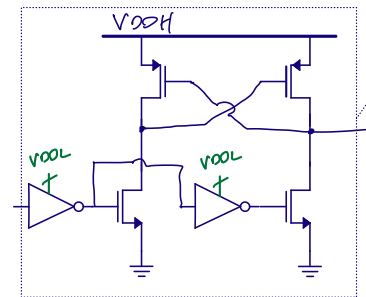
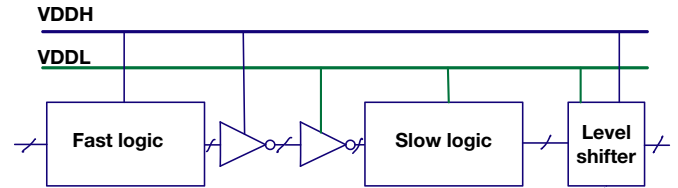
C. Reduce frequency



D. Turn off power supply ⁵



1) Reduce power supply:



2) Energy-Delay Product:

$$EDP = k \frac{C^2 V_{DD}^3}{(V_{DD} - V_t)^{1 \text{ to } 2}}$$

Differentiating with respect to

$$V_{DD}$$

⁵Often called *power gating*

and setting the result to

0

it's possible to work out that

$$V_{DD-opt} = \frac{3}{3 - 1 \text{ to } 2} V_t \in [1.5, 3] V_t$$

XL. WIRES

XLI. WIRE GEOMETRY

Pitch = w + s

Aspect ratio (AR) = t/w

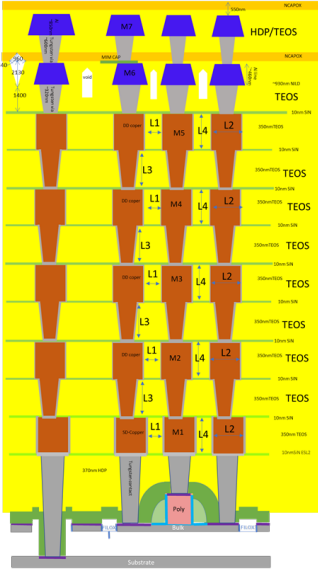
These days

$$AR \approx 2$$

XLII. METAL STACK

Often 5 - 10 layers of metal

Metal	Material	Thickness	Purpose
Metal 1	Copper	Thin	in gate routing
Metal 3 - 5	Copper	Thicker	Between gates routing
RDL	Aluminium	Ultra tick	Can tolerate high forces during wire bonding.



XLVIII. CONTACTS

Contacts and vias can have 2-20

Ω

Must use many contacts/vias for high current wires

XLIX. WIRE CAPACITANCE

Dense wires has about

$0.2\text{ fF}/\mu\text{m}$

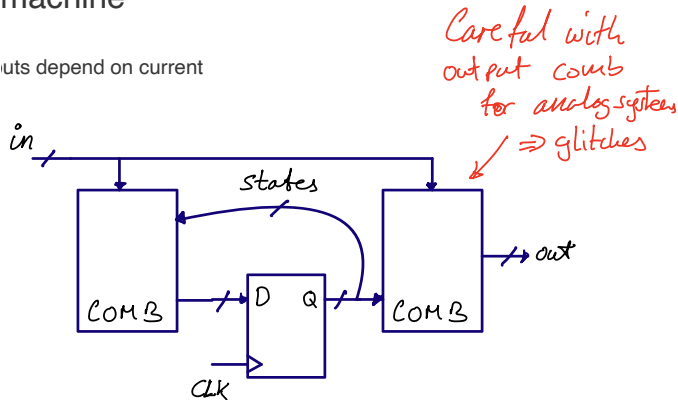
L. FSM

LI. MEALY MACHINE

An FSM where outputs depend on current state and inputs

machine

uts depend on current

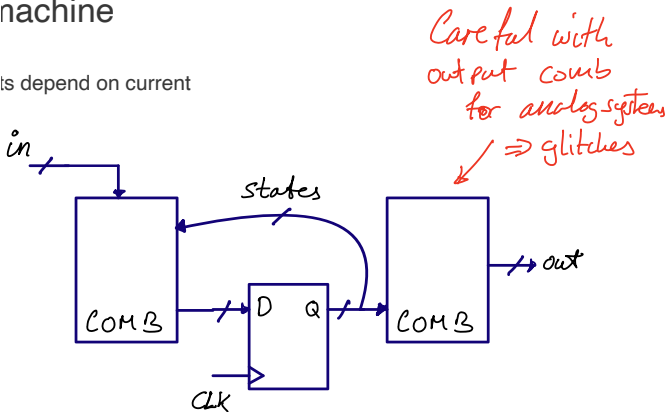


LII. MOORE MACHINE

An FSM where outputs depend on current state

nachine

ts depend on current



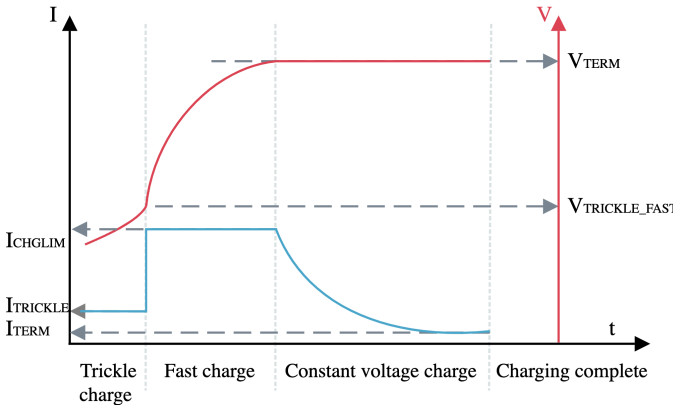
LIII. MEALY VERSUS MOORE

Parameter	Mealy	Moore
Outputs	depend on input and current state	output depend on current state
States	Same, or fewer states than Moore	
Inputs	React faster to inputs	Next clock cycle
Outputs	Can be asynchronous	Synchronous
States	Generally requires fewer states for synthesis	More states than Mealy
Counter	A counter is not a mealy machine	A counter is a Moore machine
Design	Can be tricky to design	Easy

A. dicex/sim/counter_sv/counter.v

```
module counter(  
    output logic [WIDTH-1:0] out,  
    input logic clk,  
    input logic reset  
);  
  
parameter WIDTH = 8;  
logic [WIDTH-1:0] count;  
  
always_comb begin  
    count = out + 1;  
end  
  
always_ff @(posedge clk or posedge reset) begin  
    if (reset)  
        out <= 0;  
    else  
        out <= count;  
    end  
end  
endmodule // counter
```

LIV. BATTERY CHARGER FSM



A. Li-Ion batteries

Most Li-Ion batteries can tolerate 1 C during fast charge

For Biltema 18650 cells:

$1\text{ C} = 2950\text{ mA}$

$0.1\text{ C} = 295\text{ mA}$

Most Li-Ion need to be charged to a termination voltage of 4.2 V



Too high termination voltage, or too high charging current can cause growth of lithium dendrites, that short + and -. Will end in flames. Always check manufacturer datasheet for charging curves and voltages

B. Battery charger - Inputs

Voltage above

$$V_{TRICKLE}$$

Voltage close to

$$V_{TERM}$$

If voltage close to

$$V_{TERM}$$

and current is close to

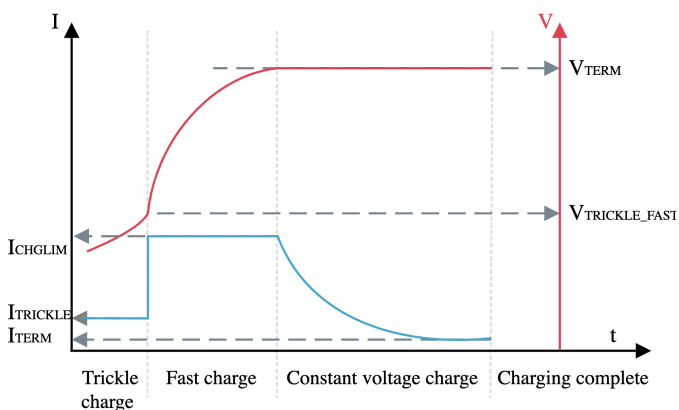
$$I_{TERM}$$

, then charging complete

If charging complete, and voltage has dropped (

$$V_{RECHARGE}$$

), then start again



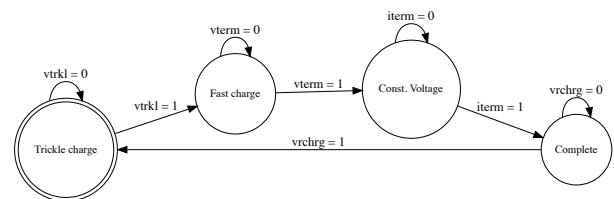
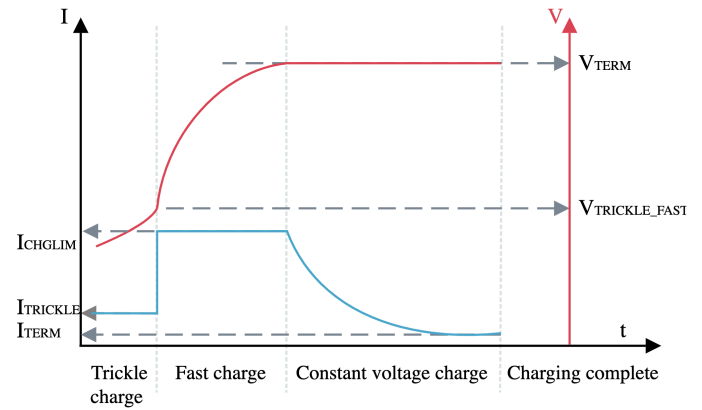
C. Battery charger - States

Trickle charge (0.1 C)

Fast charge (1 C)

Constant voltage

Charging complete



1) One way to draw FSMs - Graphviz:

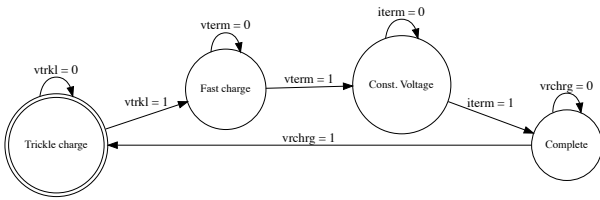
```
digraph finite_state_machine {
    rankdir=LR;
    size="8, 5"
```

```
node [shape = doublecircle, label="Trickle charge"];
node [shape = circle, label="Fast charge", fillcolor="#ffff00"];
node [shape = circle, label="Const. Voltage", fillcolor="#ffff00"];
node [shape = circle, label="Done", fillcolor="#ffff00", fontstyle=italic];
```

```
trkl -> trkl [label="vtrkl = 0"];
trkl -> fast [label="vtrkl = 1"];
fast -> fast [label="vterm = 0"];
fast -> vconst [label="vterm = 1"];
vconst -> vconst [label="item = 0"];
vconst -> done [label="item = 1"];
done -> done [label="vrchrg = 0"];
done -> trkl [label="vrchrg = 1"];
```

```
}
```

```
dot -Tpdf bcharger.dot -o bcharger.pdf
```



```

module bcharger( output logic trkl,
  output logic fast,
  output logic vconst,
  output logic done,
  input logic vtrkl,
  input logic vterm,
  input logic iterm,
  input logic vrchrg,
  input logic clk,
  input logic reset
);

parameter TRLK = 0, FAST = 1, VCONST = 2, DONE=3;
logic [1:0] state;
logic [1:0] next_state;

// Figure out the next state
always_comb begin
  case (state)
    TRLK: next_state = vtrkl ? FAST : TRLK;
    FAST: next_state = vterm ? VCONST : FAST;
    VCONST: next_state = iterm ? DONE : VCONST;
    DONE: next_state = vrchrg ? TRLK : DONE;
    default: next_state = TRLK;
  endcase // case (state)
end

// Control output signals
always_ff @(posedge clk or posedge reset) begin
  if(reset) begin
    state <= TRLK;
    trkl <= 1;
    fast <= 0;
    vconst <= 0;
    done <= 0;
  end
  else begin
    state <= next_state;
    case (state)
      TRLK: begin
        trkl <= 1;
        fast <= 0;
        vconst <= 0;
        done <= 0;
      end
      FAST: begin
        trkl <= 0;
        fast <= 1;
        vconst <= 0;
        done <= 0;
      end
      VCONST: begin
        trkl <= 0;
        fast <= 0;
        vconst <= 1;
        done <= 0;
      end
      DONE: begin
        trkl <= 0;
        fast <= 0;
        vconst <= 0;
        done <= 1;
      end
    endcase // case (state)
  end // else: !if(reset)
end
endmodule

```

2) Synthesize *FSM* with yosys:
 dicex/sim/verilog/bcharger_sv/bcharger.ys

```

# read design
read_verilog -sv bcharger.sv;
hierarchy -top bcharger;

# the high-level stuff
fsm; opt; memory; opt;

# mapping to internal cell library
techmap; opt;
synth;
opt_clean;

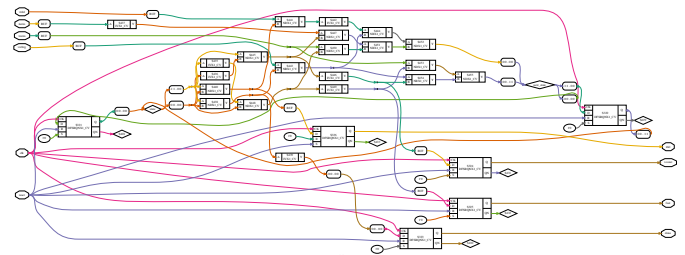
# mapping flip-flops
dfflibmap -liberty ../../lib/SUN_TR_GF130N.lib

# mapping logic
abc -liberty ../../lib/SUN_TR_GF130N.lib

# write synth netlist
write_verilog bcharger_netlist.v
read_verilog ../../lib/SUN_TR_GF130N_empty.v
write_spice -big_endian -neg AVSS -pos AVDD -top bcharger bcharger_netlist.spice

# write dot so we can make image
show -format dot -prefix bcharger_synth -colors 1 -width -stretch
clean

```



Carsten Wulff received the M.Sc. and Ph.D. degrees in electrical engineering from the Department of Electronics and Telecommunication, Norwegian University of Science and Technology (NTNU), in 2002 and 2008, respectively. During his Ph.D. work at NTNU, he worked on open-loop sigma-delta modulators and analog-to-digital converters in nanoscale CMOS technologies. In 2006-2007, he was a Visiting Researcher with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada. Since 2008 he's been with Nordic Semiconductor in various roles, from analog designer, to Wireless Group Manager, to currently Principle IC Scientist. He's also an Adjunct Associate Professor at NTNU. His present research interests includes analog and mixed-signal CMOS design, design of high-efficiency analog-to-digital converters and low-power wireless transceivers. He is the developer of Custom IC Compiler, a general purpose integrated circuit compiler.