

Masking Countermeasures in Cryptographic Hardware: Part II

Cryptographic Hardware for Embedded Systems

ECE 3170

Fall 2025

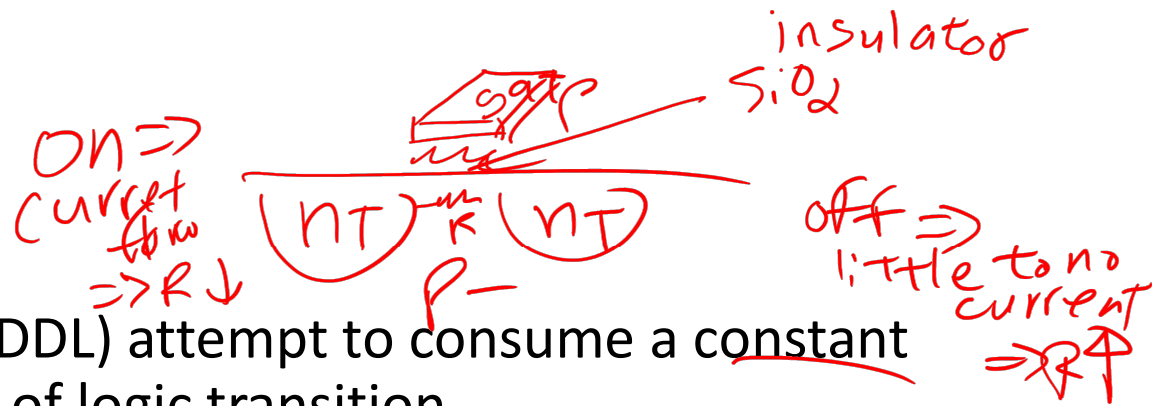
Assoc. Prof. Vincent John Mooney III

Georgia Institute of Technology

Reading

- This lecture is based on three sources:
- Chapter 9 of *Power Analysis Attacks: Revealing the Secrets of Smart Cards* by Mangard et al., 2007, ISBN-13: 978-0-387-30857-9, ISBN-10: 0-387-30857-1, e-ISBN-10: 0-387-38162-7.
- ✓ • T. Popp and S. Mangard, “Masked Dual-Rail Pre-charge Logic: Differential Power Analysis Resistance Without Routing Constraints,” Cryptographic Hardware for Embedded Systems (CHES) conference, 2005.
assigned hw 10
- Chapter 2 of *Handbook of Applied Cryptography* by Menezes et al., 1996, ISBN: 978-1-119-09672-6.

Motivation



- Prior hiding approaches (such as WDDL) attempt to consume a constant amount of energy regardless of logic transition
 - Basis: same number of wires transition from one logic value (0 or 1) to the opposite
- A major problem with prior work is that semi-custom chip design tools place and route standard cell logic gates without guaranteeing that all wires have equivalent capacitive loads State w/ L diff
 - Furthermore, there does not appear to be any straightforward manner to do so



Recall Slide 30 of Lecture 26 Hiding II

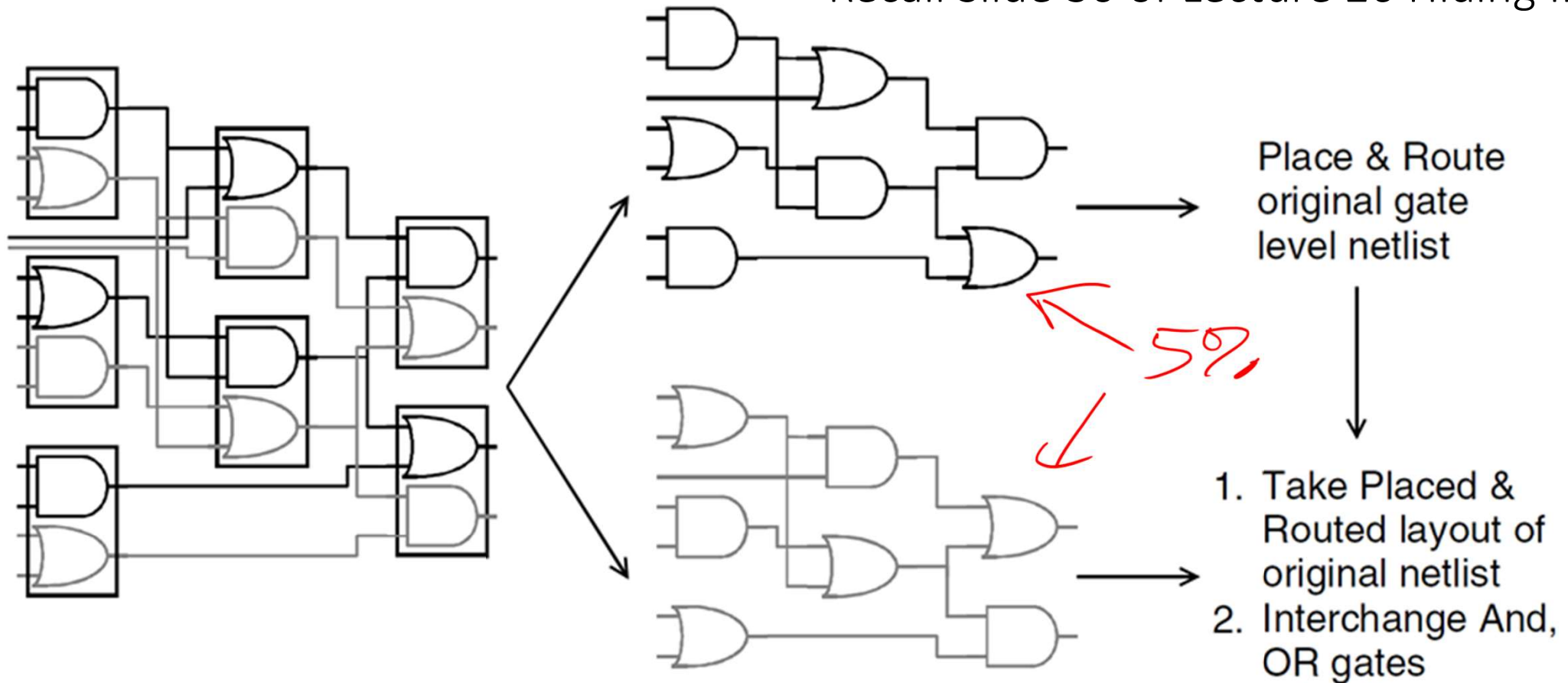


Fig. 5. Derivation of Divided WDDL

©Georgia Institute of Technology, 2018-2025

Masked Dual-rail Precharge Logic (MDPL)

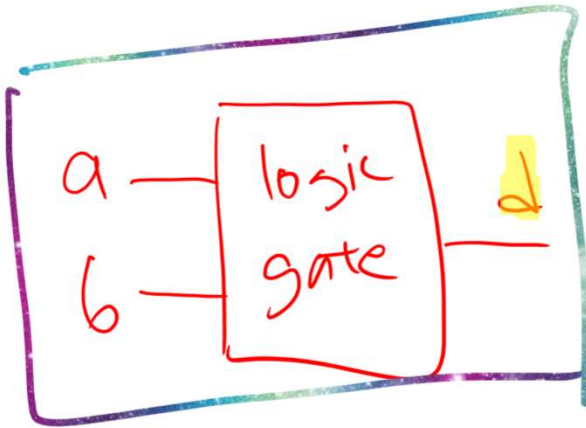
please read the
paper
(also hw 16)

mp running speckmark, embsy

Table 1. Transitions of the value d of a node in a CMOS

2-input logic gate

For example,



d_{t-1}	d_t	Energy	Probability	
0	→ 0	E_{00}	p_{00}	= 35%
0	→ 1	E_{01}	p_{01}	= 15%
1	→ 0	E_{10}	p_{10}	= 20%
1	→ 1	E_{11}	p_{11}	= 30%

d_{t-1} transition to d_t

Expected Value of the Difference of the Means

Exp. value = $\frac{\text{all occurrences of, for ex., } d_t=1}{\text{prob. of occurrence}}$

$$\mathcal{E}(DM_{d_t}) = \mathcal{E}(M_{d_t=1}) - \mathcal{E}(M_{d_t=0}) = \left(\frac{p_{11}E_{11} + p_{01}E_{01}}{p_{11} + p_{01}} \right) - \left(\frac{p_{00}E_{00} + p_{10}E_{10}}{p_{00} + p_{10}} \right)$$

if it can

mean energy consumption for case where $d_t=1$

$d_t=0$

Idea

if xor output. does not change,
gate only energy loss due to leakage

- In standard CMOS logic, $E_{00} \approx E_{11} \ll E_{10} \neq E_{01}$
- Prior ^{hiding} approaches fail to guarantee $E_{00} = E_{11} = E_{10} = E_{01}$
- Therefore, another approach is to mask the value of d
 - Instead of representing d by two wires for each polarity $\bar{d}$ and d
 - Generate a random mask m and update m every clock cycle
 - $d_m = d \oplus m$

DKNE
big
overhead

suppose
 d is part
of the key
or reveal
part of key

$$d = 0101 \quad m = 1100$$

$$d \oplus m = 1001$$

assuming adversary does not know m
 $\Rightarrow$ adversary does not know d

d_{t-1}	m_{t-1}	d_t	m_t
0	0	0	0
0	0	1	1
1	1	0	0
1	1	1	1

$$d_{m_{t-1}} = d_{t-1} \oplus m_{t-1}$$

assume:
large
sample
size
data values
and
mask
values,
uniform
distr.

Table 2. Transitions of the value d_m of a masked node

Line no.	d_{t-1}	m_{t-1}	$d_{m_{t-1}}$	d_t	m_t	d_{m_t}	Energy	Probability
1	0	0	0	0	0	0	E_{00}	$\frac{1}{4}p_{00}$
2	0	0	0	1	1	0	E_{00}	$\frac{1}{4}p_{01}$
3	1	1	0	0	0	0	E_{00}	$\frac{1}{4}p_{10}$
4	1	1	0	1	1	0	E_{00}	$\frac{1}{4}p_{11}$
5	0	0	0	0	1	1	E_{01}	$\frac{1}{4}p_{00}$
6	0	0	0	1	0	1	E_{01}	$\frac{1}{4}p_{01}$
7	1	1	0	0	1	1	E_{01}	$\frac{1}{4}p_{10}$
8	1	1	0	1	0	1	E_{01}	$\frac{1}{4}p_{11}$
9	0	1	1	0	0	0	E_{10}	$\frac{1}{4}p_{00}$
10	0	1	1	1	1	0	E_{10}	$\frac{1}{4}p_{01}$
11	1	0	1	0	0	0	E_{10}	$\frac{1}{4}p_{10}$
12	1	0	1	1	1	0	E_{10}	$\frac{1}{4}p_{11}$
13	0	1	1	0	1	1	E_{11}	$\frac{1}{4}p_{00}$
14	0	1	1	1	0	1	E_{11}	$\frac{1}{4}p_{01}$
15	1	0	1	0	1	1	E_{11}	$\frac{1}{4}p_{10}$
16	1	0	1	1	0	1	E_{11}	$\frac{1}{4}p_{11}$

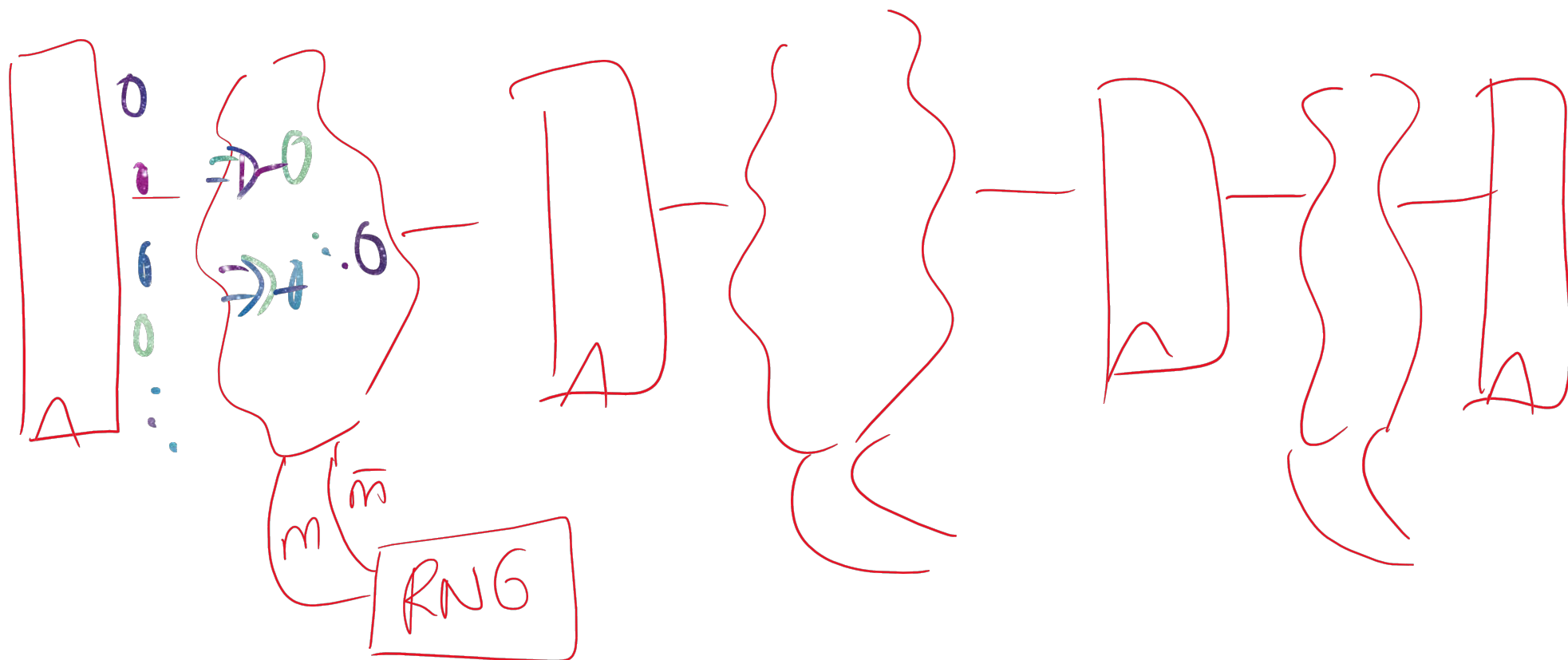
transform
 $d_{t-1} \rightarrow d_t$
to
 $d_{m_{t-1}} \rightarrow d_{m_t}$

$$\begin{aligned}
 \mathcal{E}(M_{d_t=0}) &= \frac{\text{add up rows from prev. page}}{\text{occurrence}} \\
 &= \frac{E_{00}(\frac{1}{4}p_{00}) + E_{00}(\frac{1}{4}p_{10}) + E_{01}(\frac{1}{4}(p_{00}+p_{10})) + (E_{10}+E_{11})\frac{1}{4}(p_{00}+p_{10})}{p_{00}+p_{10}}
 \end{aligned}$$

$$\mathcal{E}(M_{d_t=0}) = \mathcal{E}(M_{d_t=1}) = \left[\frac{1}{4} (E_{00} + E_{01} + E_{10} + E_{11}) \right]$$

$$\mathcal{E}(M_{d_t=1}) = \frac{(E_{00} + E_{01} + E_{10} + E_{11}) \frac{1}{4} (\cancel{p_{00}} + \cancel{p_{11}})}{\cancel{p_{00}} + \cancel{p_{11}}}$$

$$\Rightarrow \mathcal{E}(M_{d_t=0}) - \mathcal{E}(M_{d_t=1}) = 0 \quad \text{for larger sample sizes}$$



recall: $a \oplus 0 = a$ $m \oplus m = 0$
 $c = d \oplus e \Rightarrow c \oplus e = d \oplus e \oplus e = d \oplus 0 = d$

Masked Dual-rail Pre-charge Logic: MDPL

within the same clock cycle

- Every signal masked with same random value m

AND gate in MDPL

- Consider inputs a_m and b_m

- Recall $a_m = a \oplus m$ and $b_m = b \oplus m$

- Also $a = a_m \oplus m$ and $b = b_m \oplus m$

- $q = ab \Rightarrow q_m = q \oplus m = ab \oplus m$

- $\Rightarrow q_m = (a)(b) \oplus m$

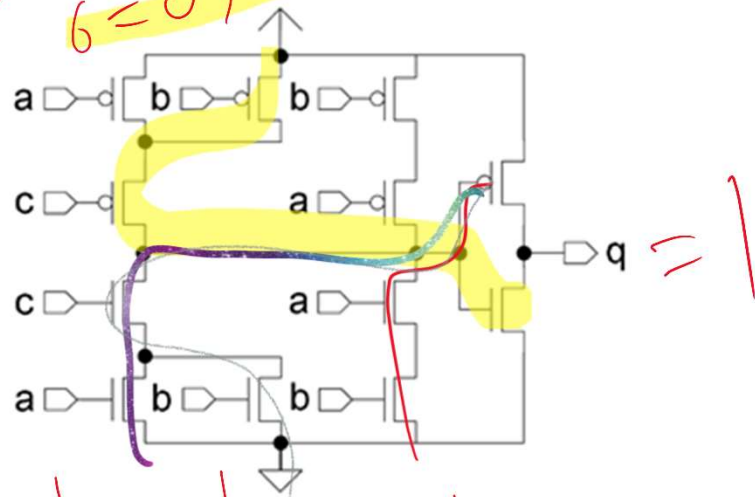
- $\Rightarrow q_m = (a_m \oplus m)(b_m \oplus m) \oplus m$

- Similarly, $\overline{q_m} = (\overline{a_m} \oplus \overline{m})(\overline{b_m} \oplus \overline{m}) \oplus \overline{m}$

Table 3. Truth table of an MDPL AND gate

Line no.	a_m	b_m	m	q_m	$\overline{a_m}$	$\overline{b_m}$	$\overline{m}$	$\overline{q_m}$
1	0	0	0	0	1	1	1	1
2	0	0	1	0	1	1	0	1
3	0	1	0	0	1	0	1	1
4	0	1	1	1	1	0	0	0
5	1	0	0	0	0	1	1	1
6	1	0	1	1	0	1	0	0
7	1	1	0	1	0	0	1	0
8	1	1	1	1	0	0	0	0

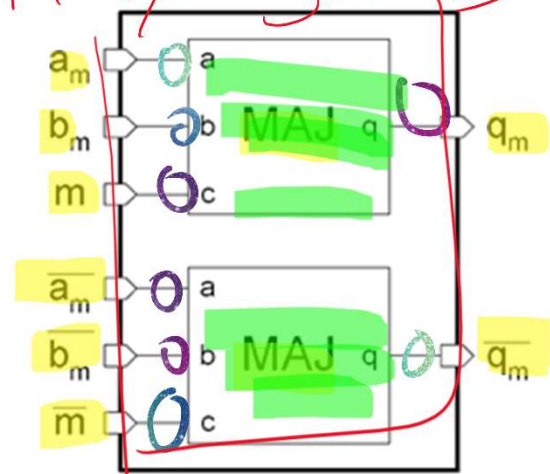
$a=0, b=0$
 $a=0, c=0$
 $b=0, c=0$



$a=1, b=1$
 $a=1, c=1$
 $b=1, c=1$

Fig. 1. Schematic of a CMOS majority gate

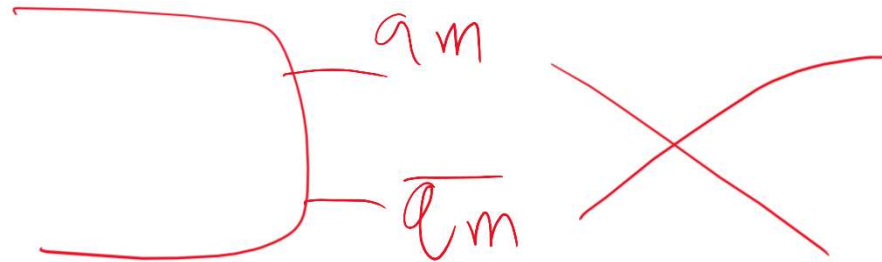
All std. cell libraries have full adders, majority gates,



0 1
 1 0

Fig. 2. Schematic of an MDPL AND gate

MDPL Operation



- As with WDDL, inversion is implemented by switching wires
- An MDPL gate for OR can also be used (not shown, see reference)
- Precharge wave sets all values to zero
 - Note that for $a_m, \bar{a}_m, b_m, \bar{b}_m, m$ and $\bar{m}$ all equal to zero makes q_m and $\bar{q}_m$ zero
- During the evaluation phase, the majority gates each make at most one transition (from 0 to 1 since they are all precharged to zero!)
- Majority gates are available in all standard cell libraries known

hazard/
glitch
free



XOR MDPL = Two MDPL AND gate plus an MDPL OR gate

Do-Do-

Table 5. MDPL cells and their CMOS implementations

assuming wires do not contribute

MDPL cell	CMOS implementation of MDPL cell	Area (gate equivalents) of		Ratio $\frac{MDPL}{CMOS}$
		MDPL cell	std. CMOS cell	
Inverter	Wire swapping	0	0.67	0
Buffer	2×Buffer	2	1	2
AND, OR (2-in)	2×MAJ (3-in)	4	1.67	2.4
NAND, NOR (2-in)	2×MAJ (3-in)	4	1	4
XOR (2-in)	6×MAJ (3-in)	12	2.33	5.1
XNOR (2-in)	6×MAJ (3-in)	12	2	6
D-Flip-Flop	2×AND, 2×OR (both 2-in)			
	2×MAJ (3-in), 1×D-FF	17.67	5	3.5

not covered

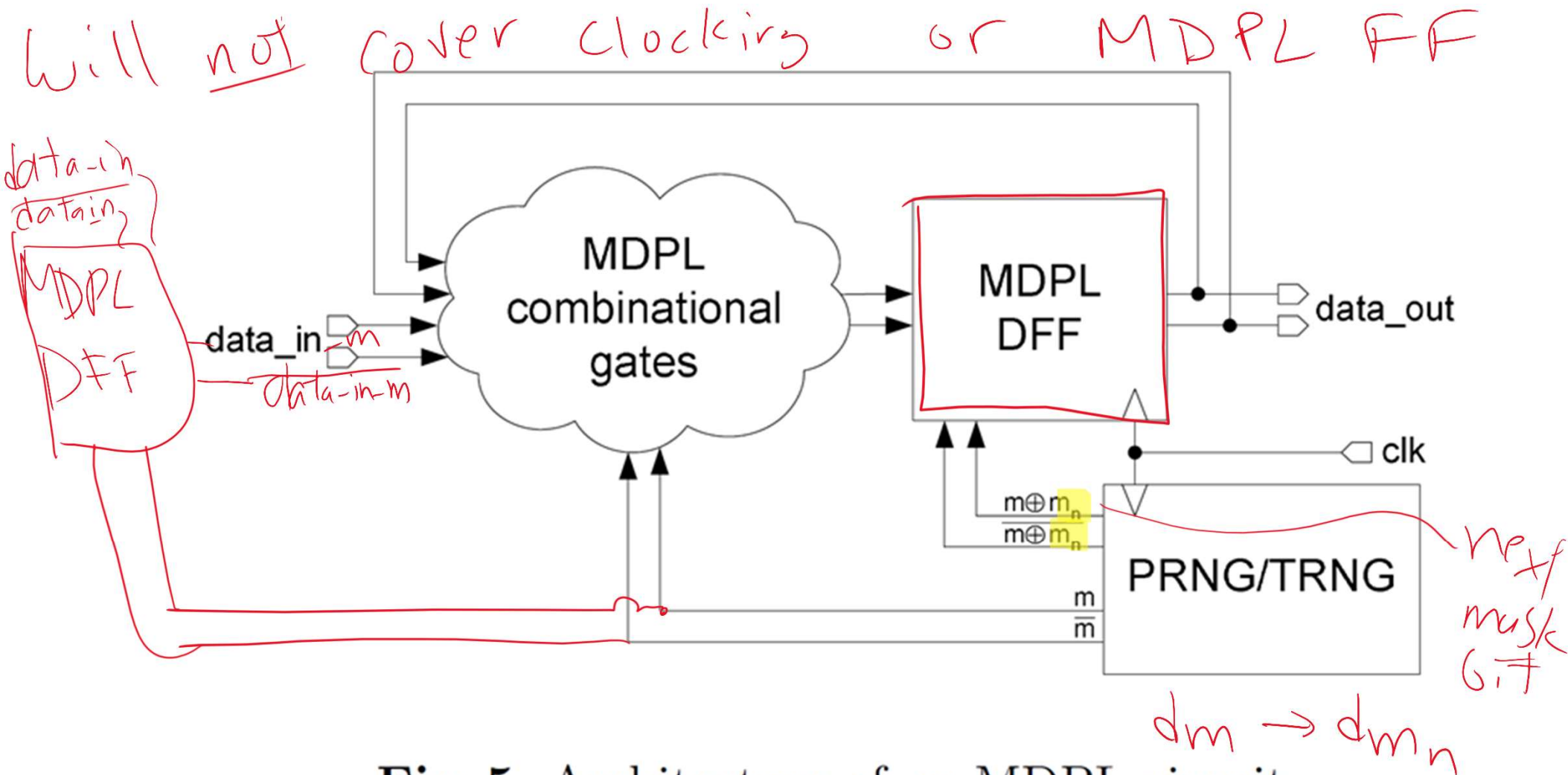


Fig. 5. Architecture of an MDPL circuit

RECALL: Hiding II Slide 30!

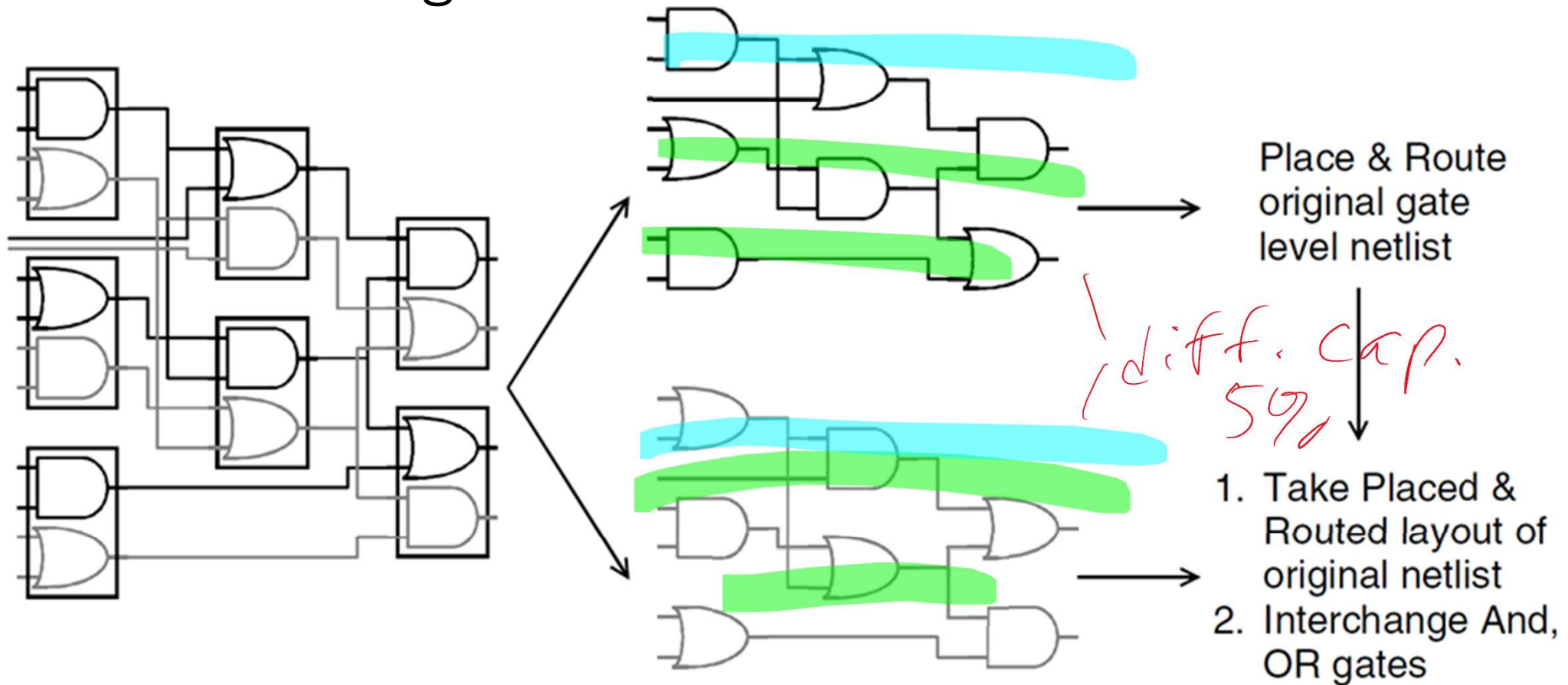


Fig. 5. Derivation of Divided WDDL

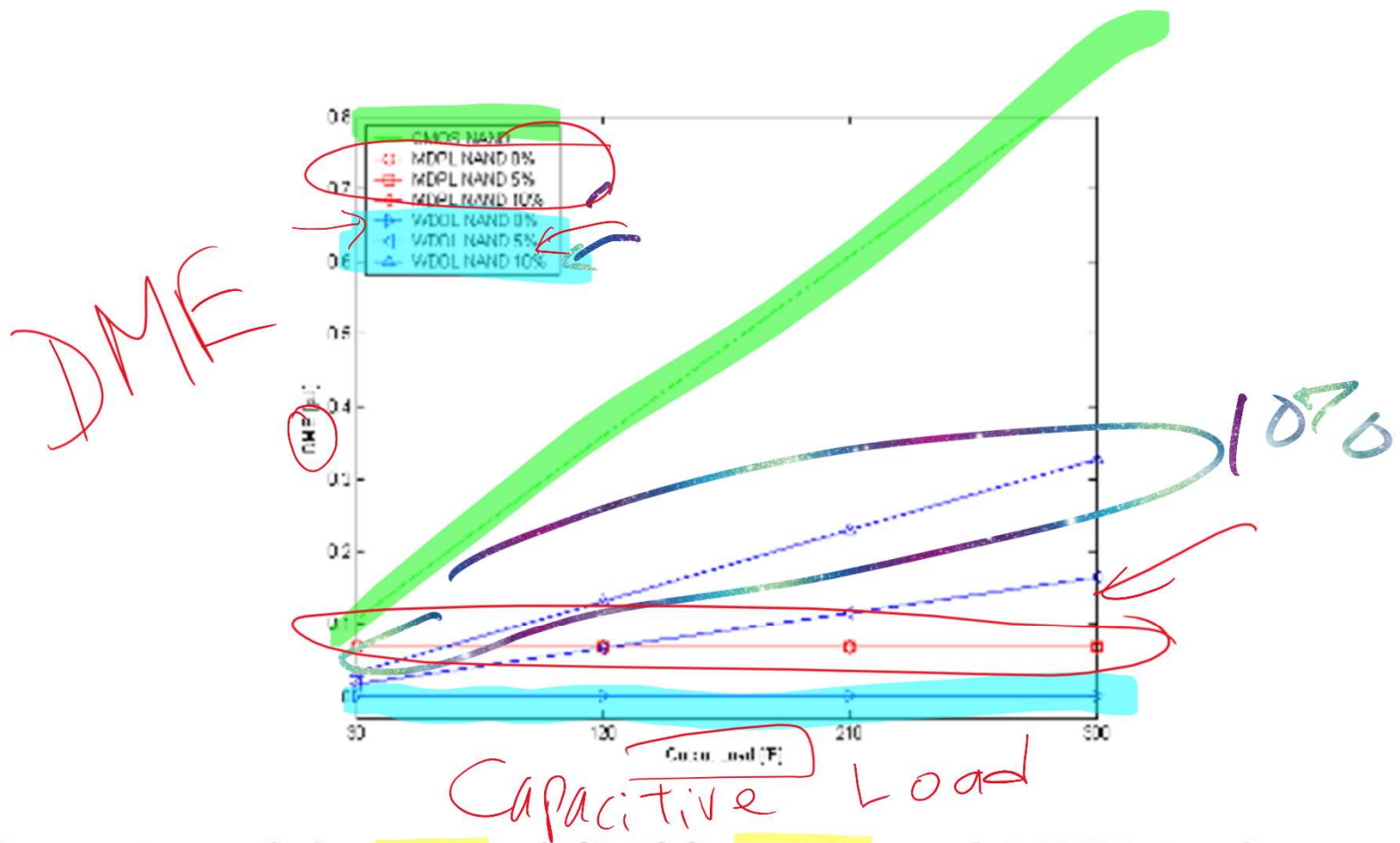


Fig. 6. Comparison of the **DME** of CMOS, **WDDL** and MDPL implementations NAND gate

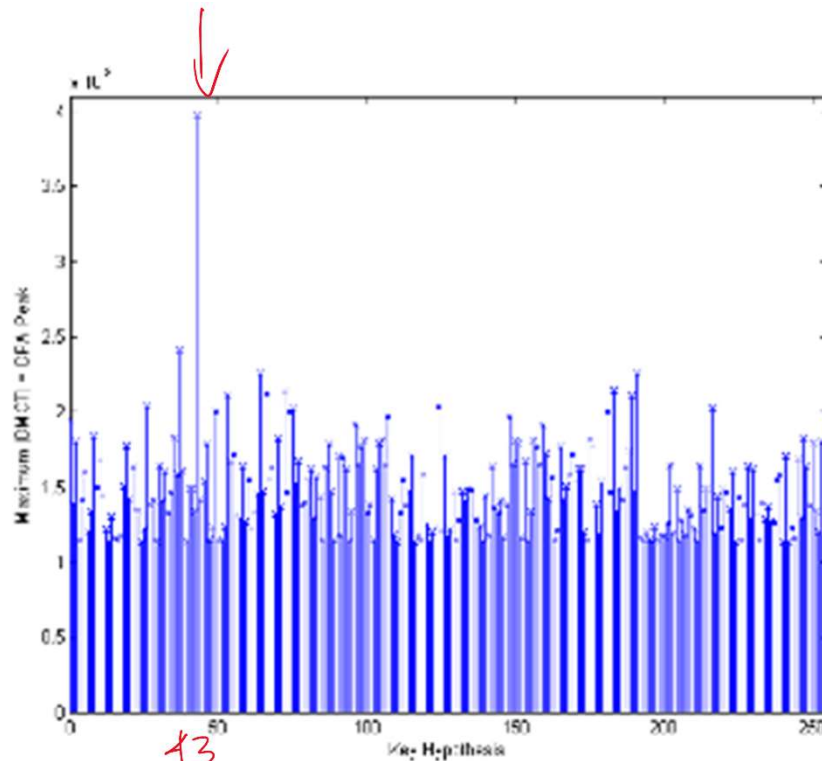


Fig. 7. DPA peaks for all key hypothesis for a DPA attack on an AES implemented in CMOS

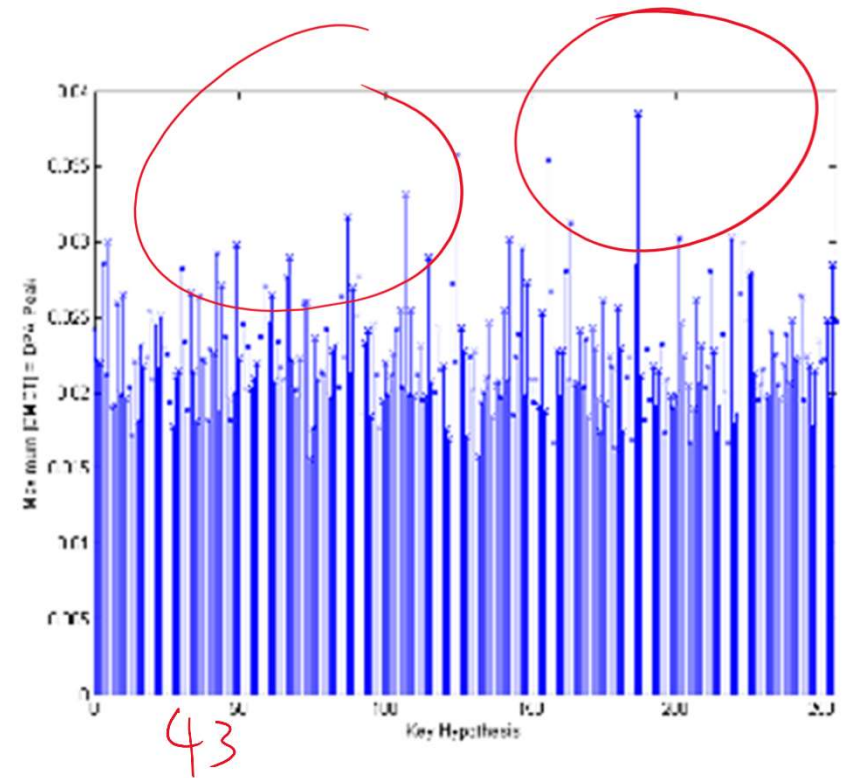


Fig. 8. DPA peaks for all key hypothesis for a DPA attack on an AES implemented in MDPL

Table 6. Comparison of AES implementations in CMOS and MDPL

	CMOS	MDPL	Ratio $\frac{MDPL}{CMOS}$
Area (gate equivalents)	3628	16465	4.54
Speed (MHz; worst-case speed corner)	16.91	9.82	0.58

$\sim 1.5 \times \uparrow$ area
55% speed $\downarrow$

Conclusion MDPL achieves nearly
flat energy consumption
does uniform RNG

Area $\uparrow$ 4.5X

clock $\downarrow$ 2X

Requires high quality RNG every clock cycle