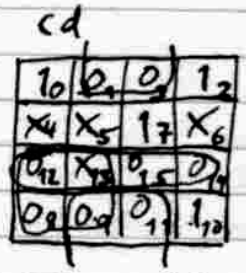
Robert Dick, EECS 303, Fall '07 Midterm

1)



If direct PoS is hard, inverting, covering ones, and De Morgan's is also fine.

$(\bar{a}+\bar{b})(b+\bar{d})(c+\bar{a})$

2) Two-level implementations can be expensive in terms of area and power for some functions. For example parity requires area exponential in inputs when implemented in two-level CMOS.

3) Exponential because the number of starting cubes may be exponential in the number of inputs. That doesn't imply exponential in minterms.

4) Need to determine whether $\{\bar{a}\bar{c}, c\bar{d}\}$ cover each of $\{\bar{c}d, a\bar{c}, a\bar{d}\}$.

Cube set: $\begin{matrix} 0X0X \\ XX10 \end{matrix}$

$\begin{pmatrix} 0X0X \\ XX10 \end{pmatrix}_{XX01} \rightarrow 0XXX$ Not a tautology.

$\begin{pmatrix} 0X0X \\ XX10 \end{pmatrix}_{1X0X} \rightarrow \emptyset$ Not a tautology.

$\begin{pmatrix} 0X0X \\ XX10 \end{pmatrix}_{1XX0} \rightarrow XX1X$ Not a tautology.

all of the redundant cubes are "partially redundant."

5) A unate cover is a tautology if and only if it contains a cube that covers the entire input space. This is true because all cubes in a unate cover intersect at at least one point. Therefore, the only way the far corner of the space can be covered is if one of the cubes covers the entire space.

6) No. Electrons would tunnel from floating gate to substrate: they would leak away.

7) Modern FPGA families have basic logic units with 4-6 inputs.

8) a) Dynamically-configurable but volatile.
b) Non-volatile but two-level.
c) Dense but does not benefit from function minimization.
d) Restore signals but cannot have High-Z outputs.
e) Support High-Z outputs but do not restore signals, which can prevent deep networks and complicate timing analysis.

9)

Number	Encoding
0	000
1	001
2	011
3	010
4	110
5	111
6	101
7	100

10)

Number	Encoding	Wires to switch
0	000	1
1	001	2
2	010	1
3	011	2
4	110	1
5	111	2
6	100	1
7	101	2

Average energy to switch (per cycle): 1.5e

11)

b c d	b+d	c+d	f	g	h	i
0 0 0	0	0	0	0	0	1
1 0 0	1	0	0	1	0	0
0 1 0	0	1	0	0	1	0
0 0 1	1	1	1	0	0	0

f = d
g = b
h = c
i = ...

subcircuit 1 is an encoder
subcircuit 2 is a decoder
Together, they can be used to decrease the number of wires necessary to transmit a set of values, if it is known that only one will be 1 at a time.