

1. If we set  $n$  = the number of Boolean variables, we see that  $2^{(2^n)}$  represents the pattern.

\* 1 variable :  $2^2 = 4$   
 \* 2 variables :  $2^4 = 16$

By this pattern,

\* 3 variables :  $2^8 = 64$

and in general,

\*  $n$  variables :  $2^{2^n}$

- 2.

$$w'y'z' + wxy' + w'xz + wxy$$

Wx\yz	00	01	11	10
00	1	0	0	0
01	1	1	1	0
11	1	1	1	1
10	0	0	0	0

This yields:  $xy' + xz + wxy + w'y'z'$

- 3.

$$w'x'y'z' + w'xy' + wy'z' + wxy'z + wyz'$$

Wx\yz	00	01	11	10
00	1	0	0	0
01	1	1	0	0
11	1	1	0	1
10	1	0	0	1

This yields:  $xy' + y'z' + wyz'$

- 4.

$$w'x'y'z' + w'x'y'z + w'xy'z' + w'xyz + wx'y'z' + wx'yz + wxy'z + wxyz'$$

Wx\yz	00	01	11	10
00	0	1	0	1
01	1	0	1	0
11	0	1	0	1
10	1	0	1	0

This yields: **nothing**. The grouping cannot be simplified.

5.

Parity Generator using an 8-1 Multiplexer

S <sub>0</sub>	S <sub>1</sub>	S <sub>2</sub>	Input 0 (Parity Bit)	Output
0	0	0	0	0
0	0	0	1	1

The rest of the inputs do not affect the parity of the input data. Therefore, we wire the selectors to 0,0,0 to select input 0. This will output the last bit only.

6.

$$wxy' + wxz + w'xy + xyz'$$

Wx\yz	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	1	1	1	1
10	0	0	0	0

This yields:  $wx + xy$

This requires two AND gates and one OR gate. They should be wired such that W and Y each connect to one of the AND gates and X connects to both. The two AND gates should then connect to the OR gate.

7.

$$C_3 = X_2Y_2 + (X_2 \text{ XOR } Y_2)(X_1Y_1 + (X_1 \text{ XOR } Y_1)(X_0Y_0 + (X_0 \text{ XOR } Y_0)(C_0)))$$

$$C_4 = X_3Y_3 + (X_3 \text{ XOR } Y_3)(X_2Y_2 + (X_2 \text{ XOR } Y_2)(X_1Y_1 + (X_1 \text{ XOR } Y_1)(X_0Y_0 + (X_0 \text{ XOR } Y_0)(C_0))))$$