Digital Design
Lecture 7

Combinatorial Logic

Combinatorial Logic Circuit

- General Logic Block Without Memory
  - Outputs are only a function of the current inputs
  - No storage elements to remember "state"
  - There are timing delays from input to output due to gate delays

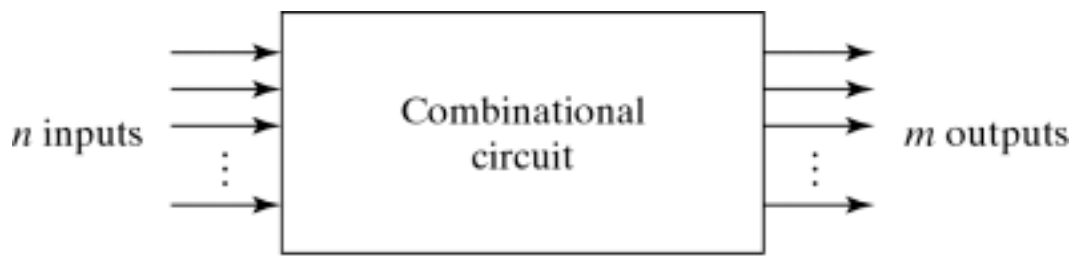


Fig. 4-1 Block Diagram of Combinational Circuit

Analysis Example

$F_2 = AB + AC + BC$

$T_1 = A + B + C$

$T_2 = ABC$

$T_3 = F_2' T_1$

$F_1 = T_3 + T_2$

Or

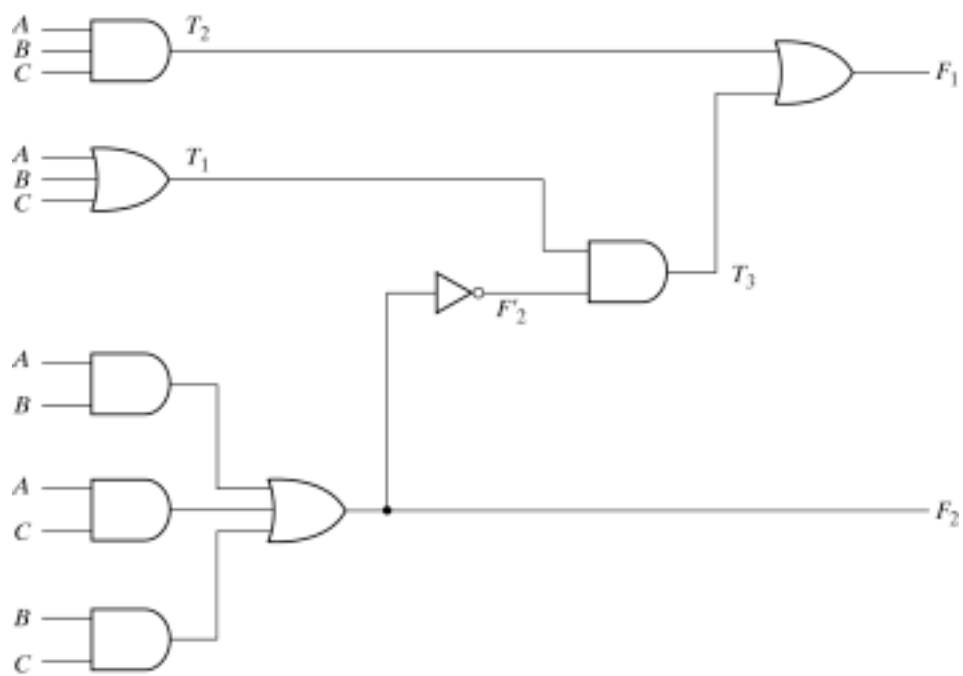


Fig. 4-2 Logic Diagram for Analysis Example

$F_1 = A'BC' + A'B'C + AB'C' + ABC$

Truth Table 4-1: Analysis Example

A	B	C	F_2	F_2´	T_1	T_2	T_3	F_1
0	0	0	0	1	0	0	0	0
0	0	1	0	1	1	0	1	1
0	1	0	0	1	1	0	1	1
0	1	1	1	0	1	0	0	0
1	0	0	0	1	1	0	1	1
1	0	1	1	0	1	0	0	0
1	1	0	1	0	1	0	0	0
1	1	1	1	0	1	1	0	1

Truth Table 4-2: Code Conversion

Input BCD				Output Excess-3 Code			
A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

Code Conversion Karnaugh Maps

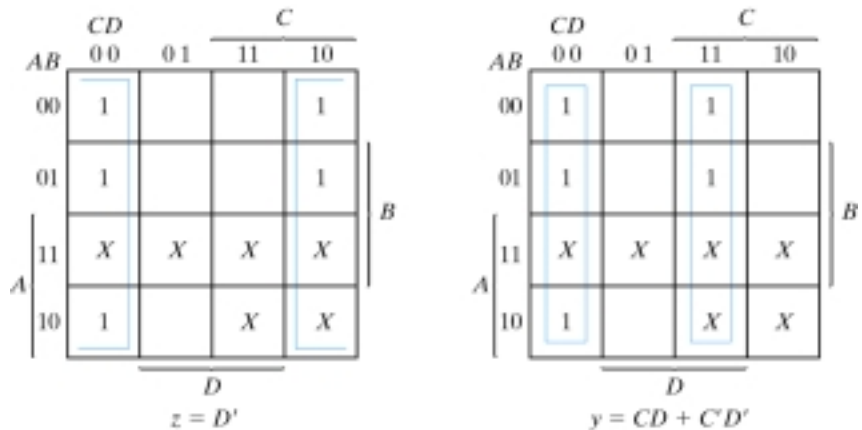


Fig. 4-3 Maps for BCD to Excess-3 Code Converter

BCD to Excess-3 Code Converter

$z = D'$

$y = CD + C'D'$

$= CD + (C+D)'$

$x = B'C + BD' + BC'D'$

$= B'(C+D) + B(C+D)'$

$w = A + BC + BD$

$= A + B(C+D)$

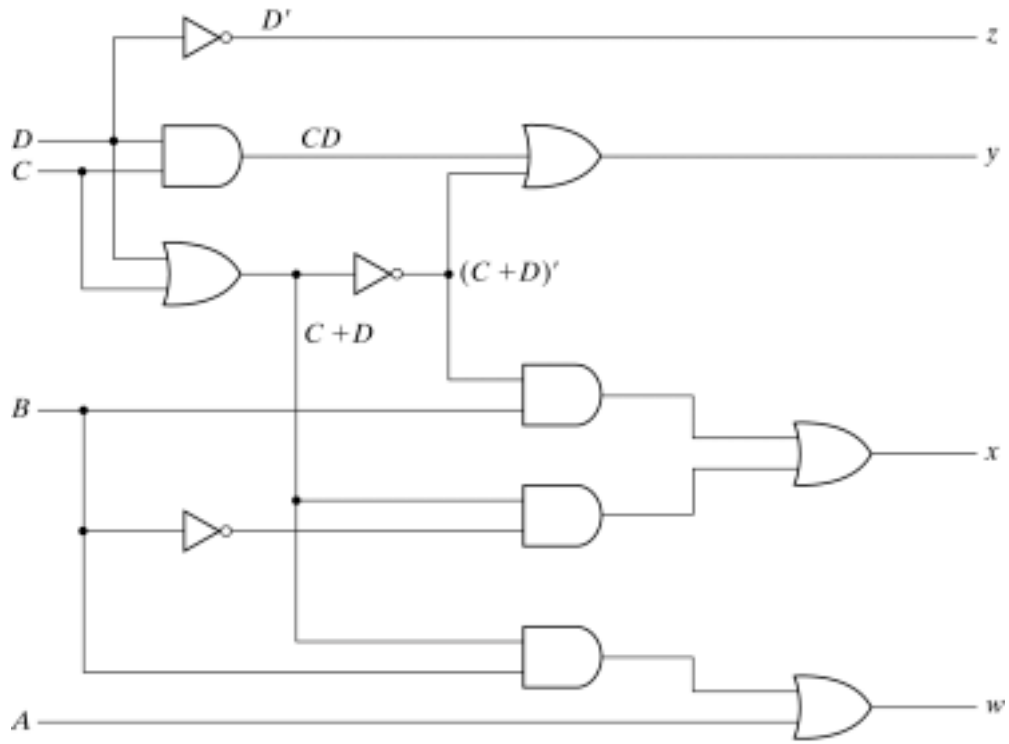
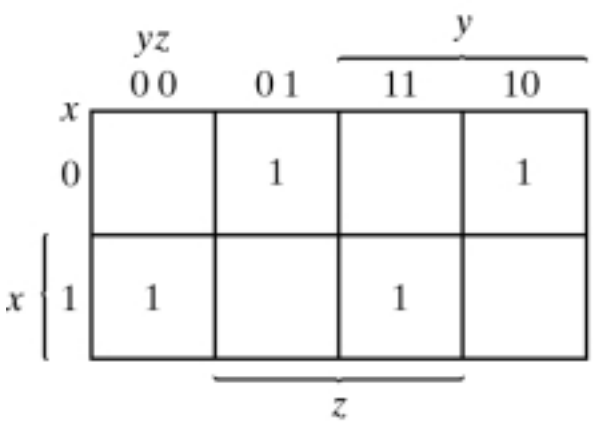


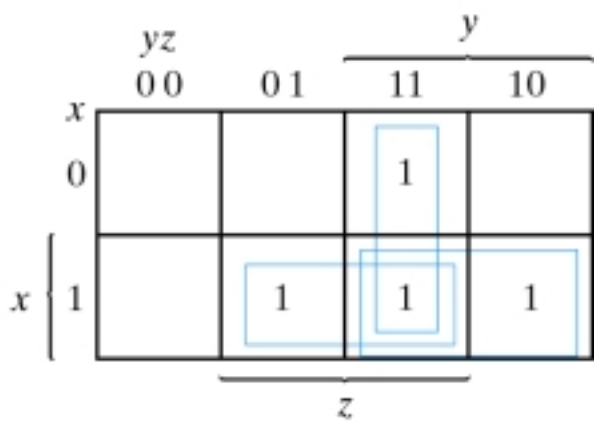
Fig. 4-4 Logic Diagram for BCD to Excess-3 Code Converter

Half Adder

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0



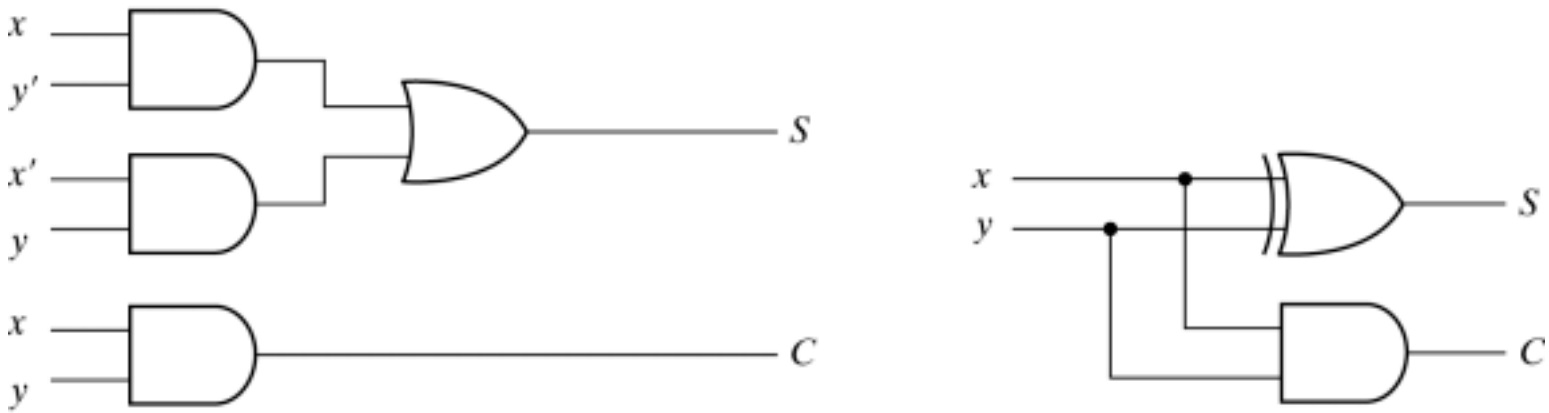
$S = x'y'z + x'yz' + xy'z' + xyz$



$S = xy + xz + yz$
$= xy + xy'z + x'yz$

Fig. 4-6 Maps for Full Adder

Half Adder (cont.)



(a) $S = xy' + x'y$
$C = xy$

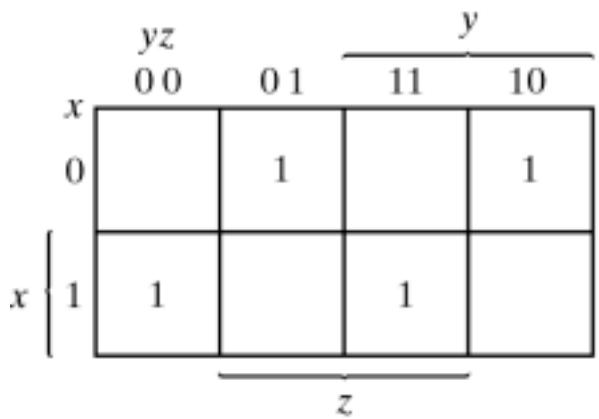
(b) $S = x \oplus y$
$C = xy$

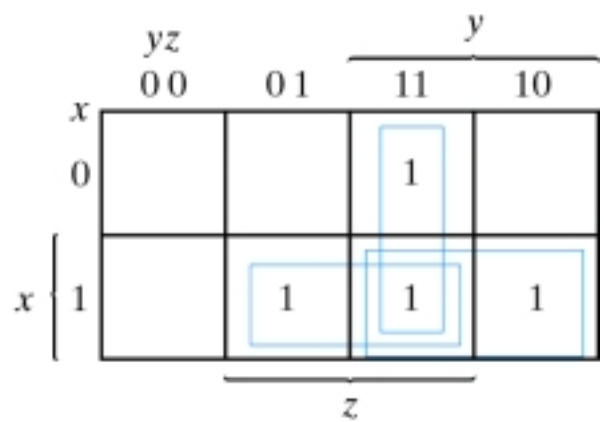
Fig. 4-5 Implementation of Half-Adder

Full Adder

-

x	y	C	S
0	0	0	0
0	0	0	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	1	0
1	1	1	1





$$S = xy + xz + yz$$
$$= xy + xy'z + x'yz$$

Full Adder: Sum of Products

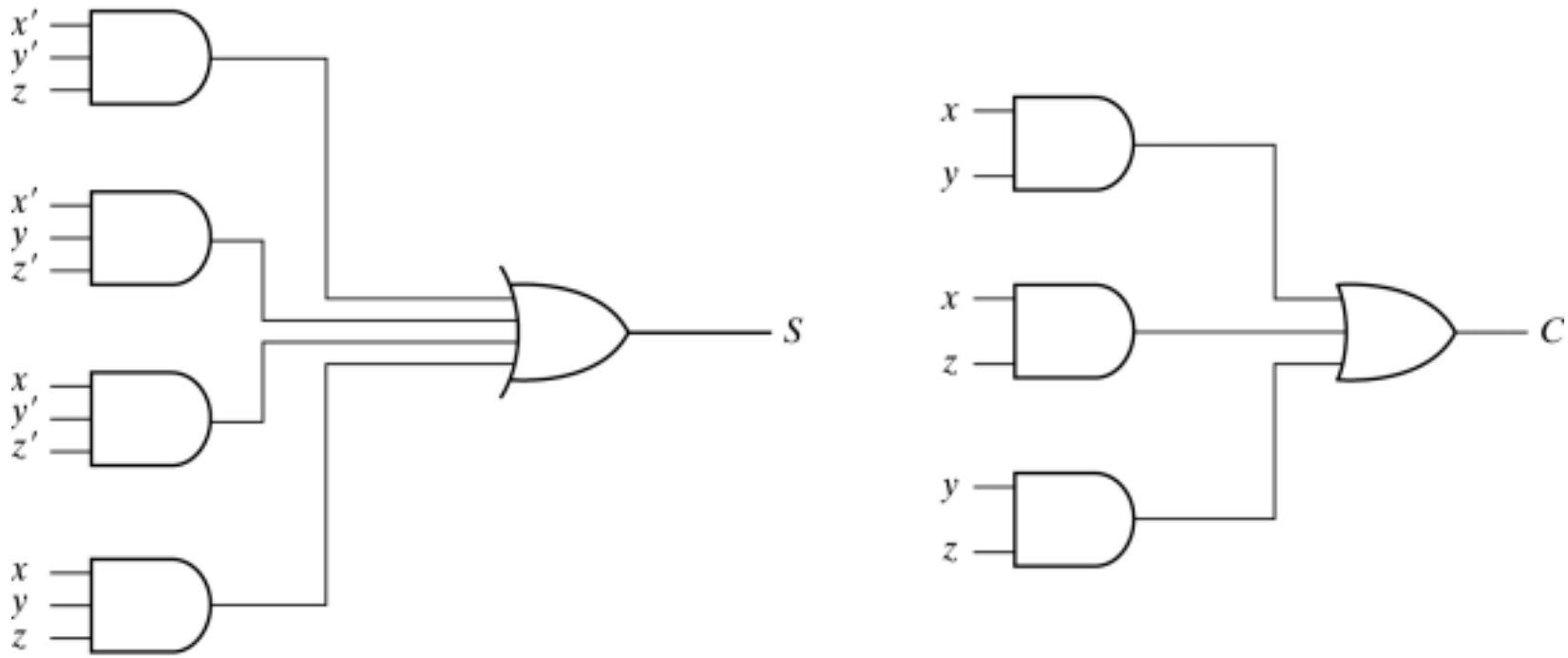


Fig. 4-7 Implementation of Full Adder in Sum of Products

Full Adder: Using Half Adders

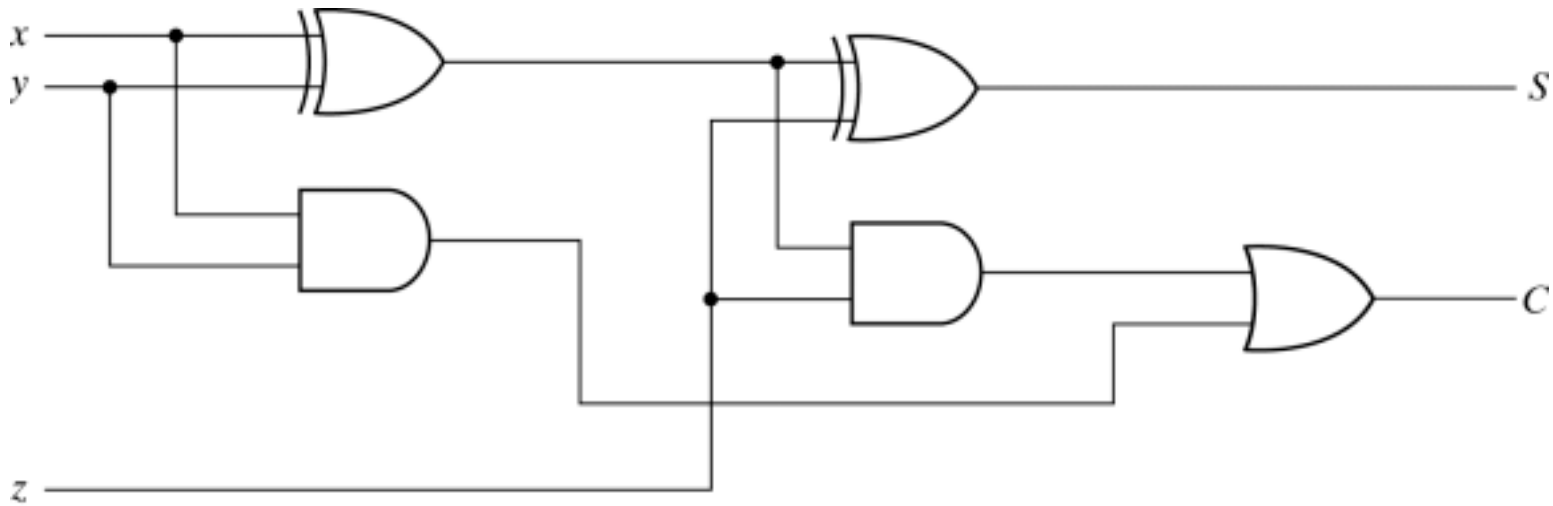


Fig. 4-8 Implementation of Full Adder with Two Half Adders and an OR Gate

4-Bit Adder

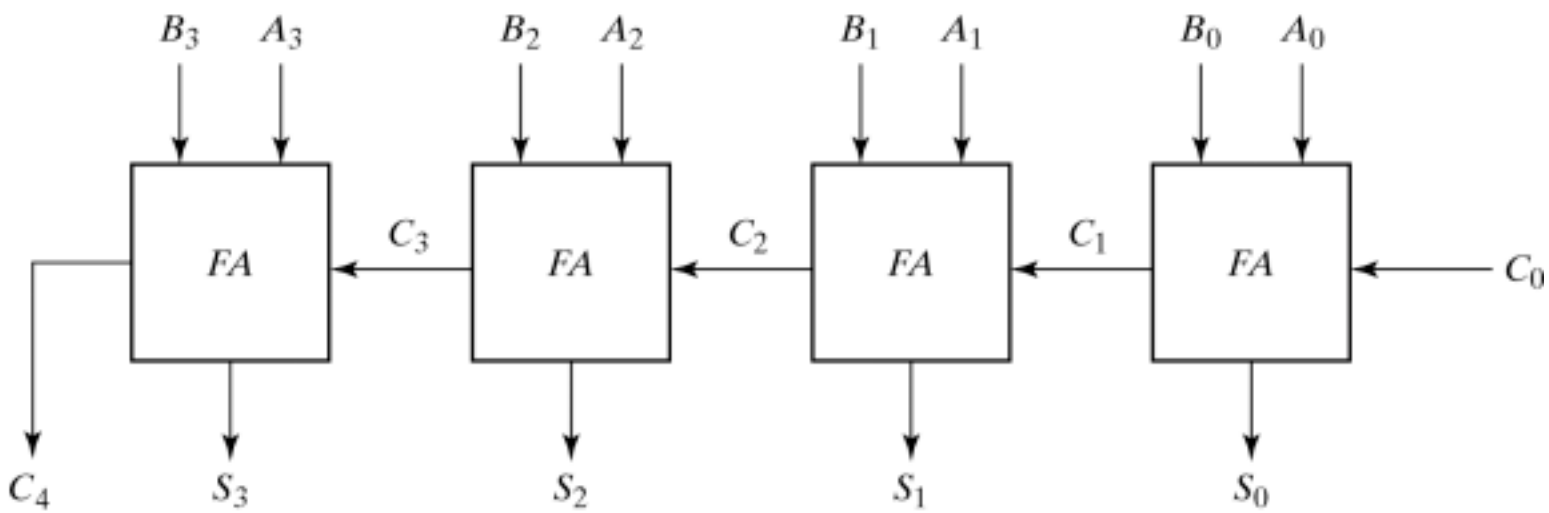


Fig. 4-9 4-Bit Adder

Carry Propagation

- Carry bits must “ripple” through each stage of a multi-bit adder before the output settles down to the correct result
  (two gate delays per bit)

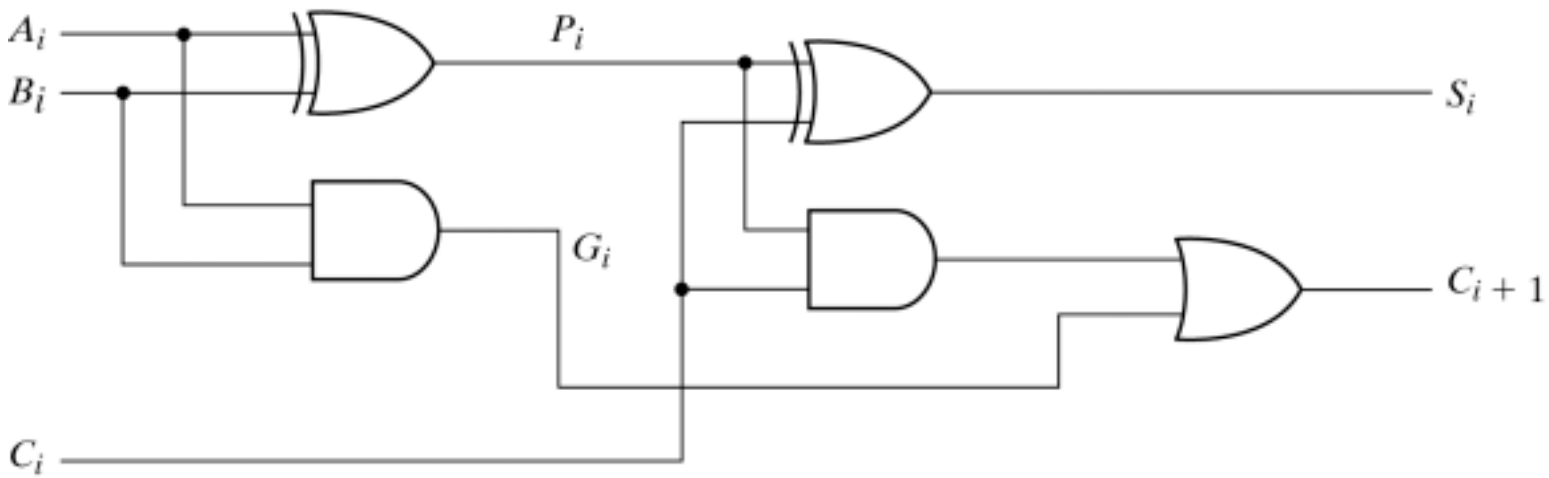


Fig. 4-10 Full Adder with P and G Shown

Carry Look Ahead

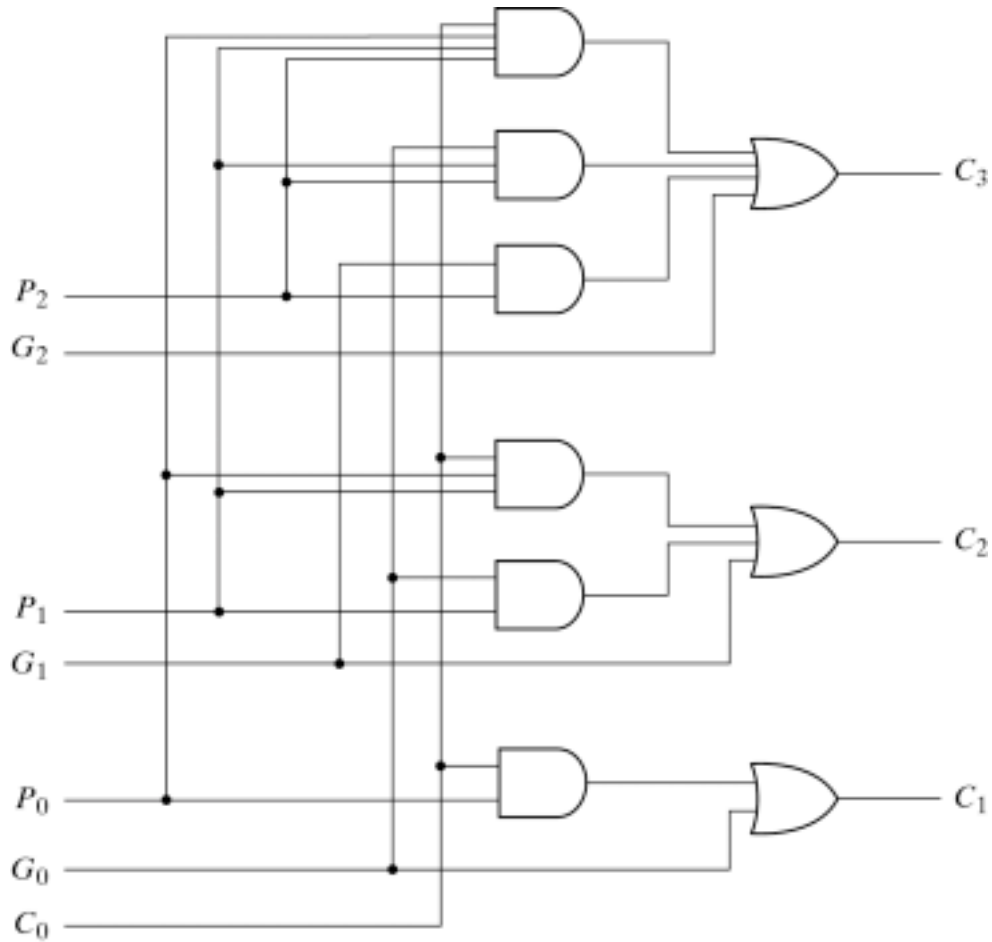


Fig. 4-11 Logic Diagram of Carry Lookahead Generator

4-Bit Adder with Carry Look Ahead

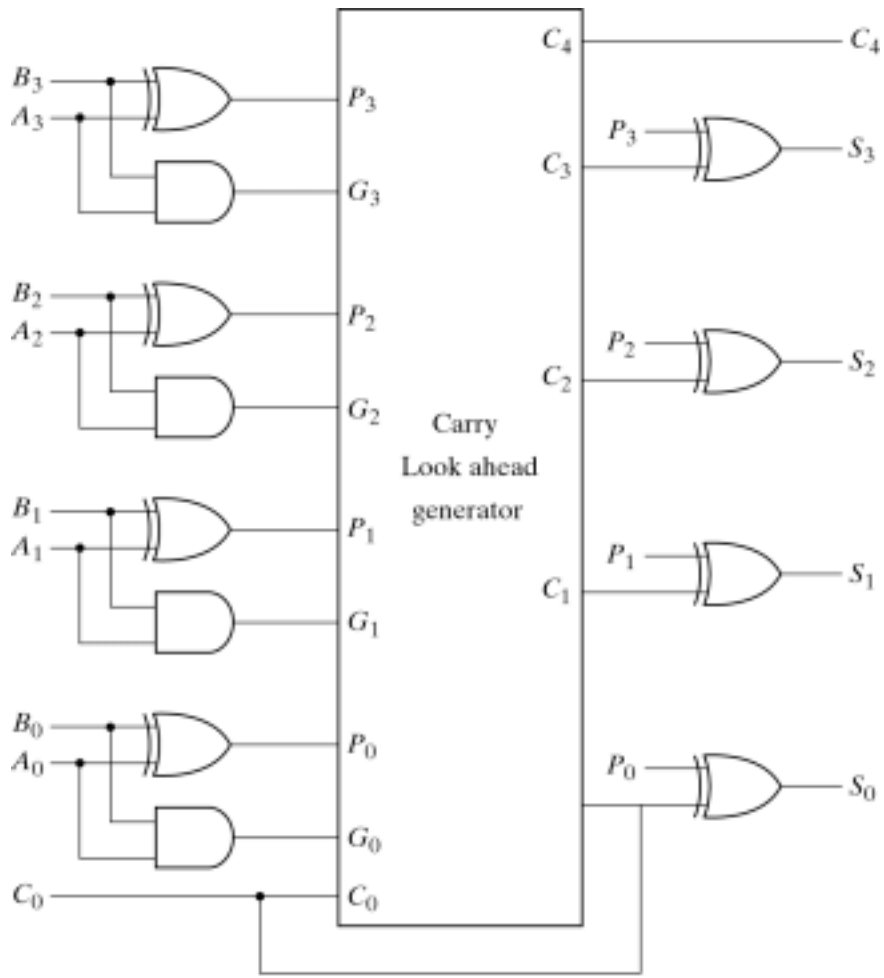


Fig. 4-12 4-Bit Adder with Carry Lookahead

Binary Subtraction

M = 1 subtract

V = overflow detect

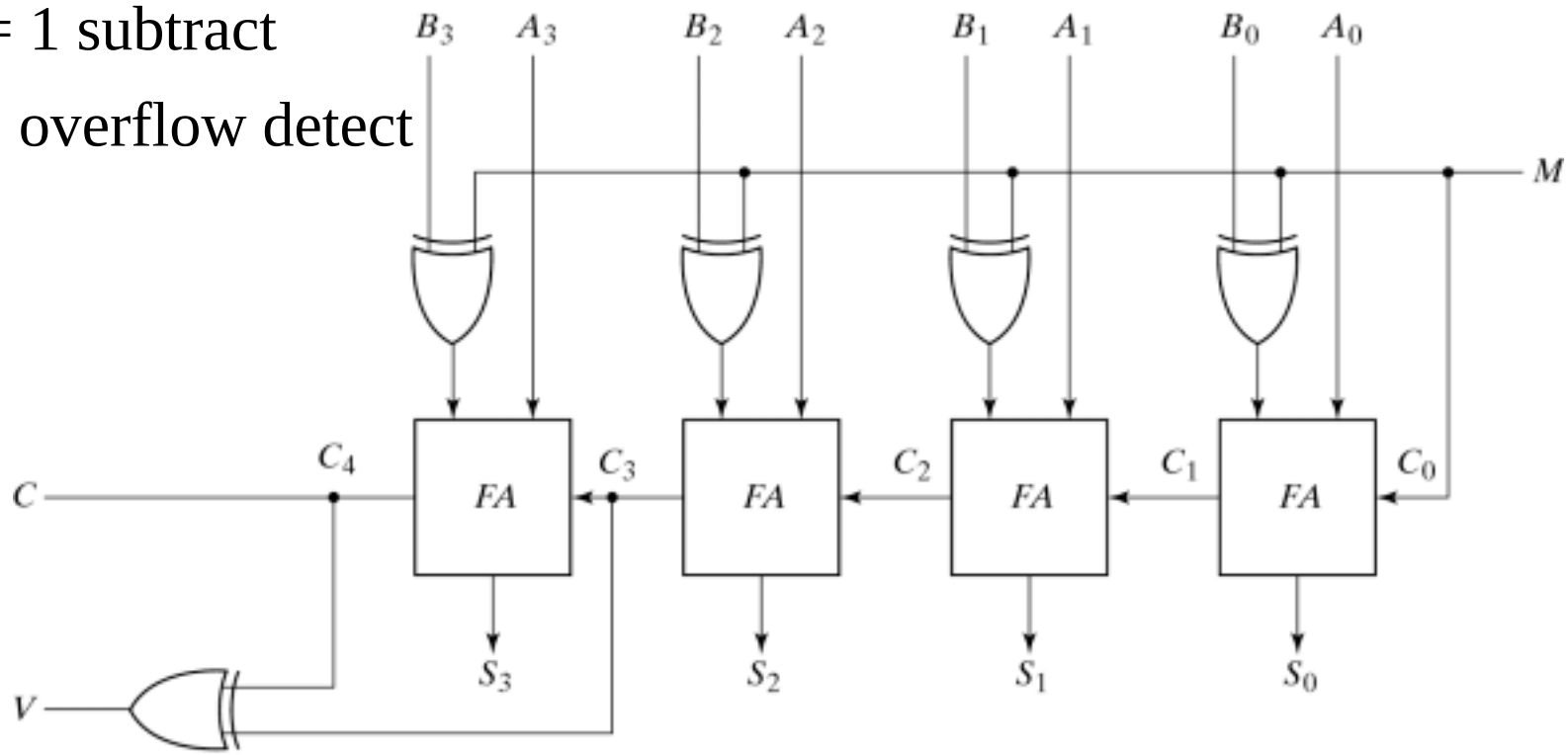


Fig. 4-13 4-Bit Adder Subtractor

BCD Adder

- Carry 0110 when first adder result overflows

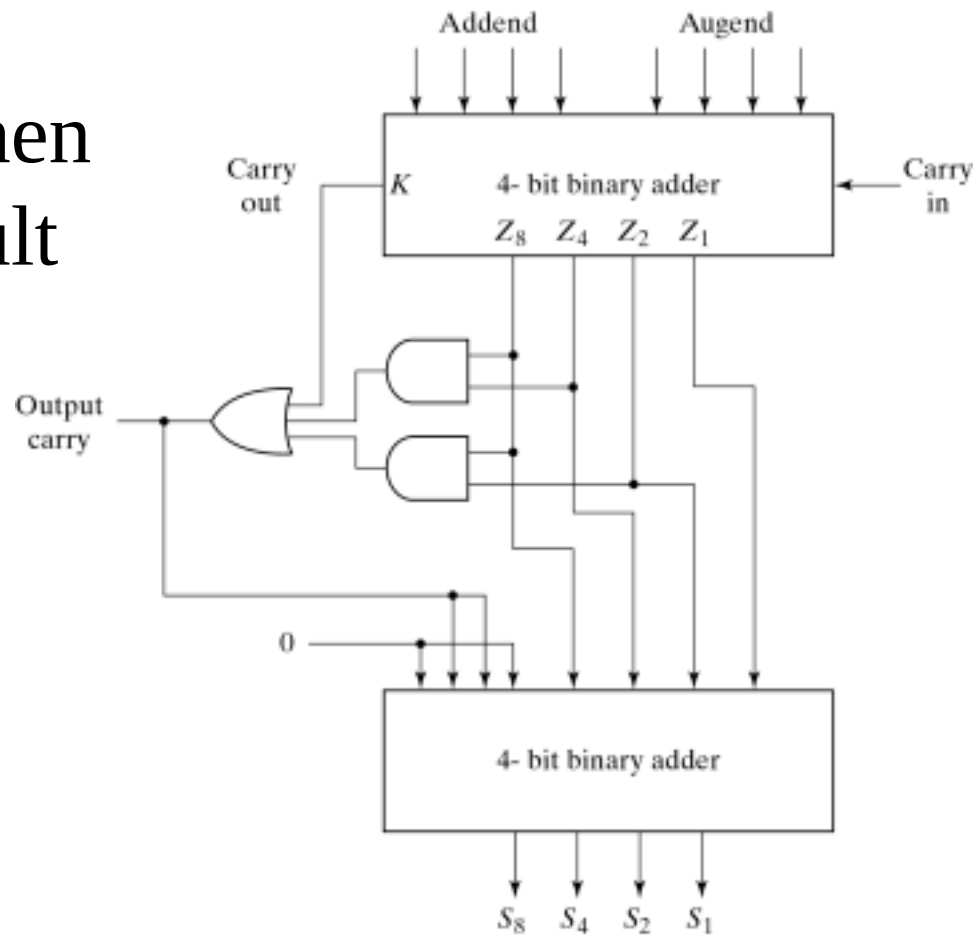


Fig. 4-14 Block Diagram of a BCD Adder