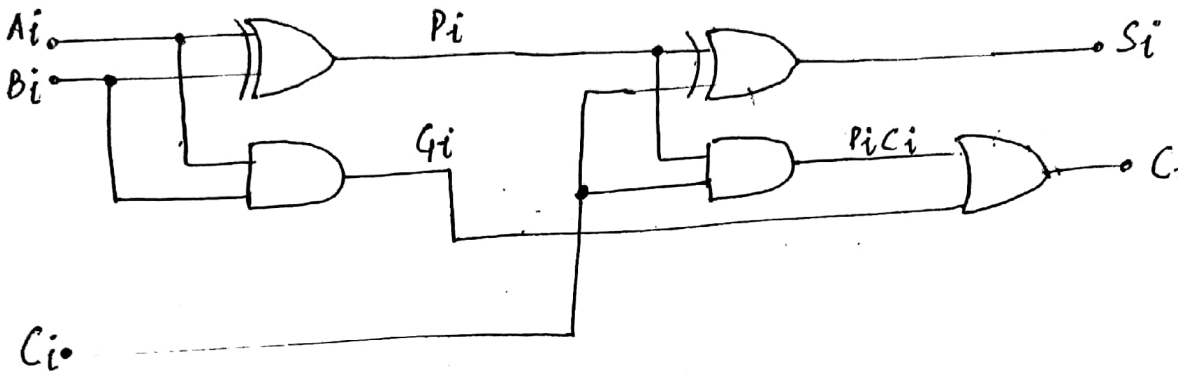
To reduce carry propagation delay time:-
- employ faster gates with reduced delays
- increase equipment complexity in such a way that the carry delay time is reduced.

Carry Lookahead is one such technique for reducing th propagation time.



Consider the ckt of full adder shown above.

$P_i \rightarrow$ carry propagate. It is the term associated with the propagation of the carry C_i to C_{i+1}

$G_i \rightarrow$ carry generate. It produces a carry of 1 wh both A_i & B_i are 1, regardless of C_i.

$$P_i = A_i \oplus B_i$$
$$G_i = A_i B_i$$
$$\therefore S_i = P_i \oplus C_i$$
$$C_{i+1} = G_i + P_i C_i$$

Writing carry outputs of each stage:

C_0 = input carry

$C_1 = G_0 + P_0 C_0$

$C_2 = G_1 + P_1 C_1 = G_1 + P_1 (G_0 + P_0 C_0)$
$= G_1 + P_1 G_0 + P_1 P_0 C_0$

$C_3 = G_2 + P_2 C_2 = G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0)$
$= G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C$

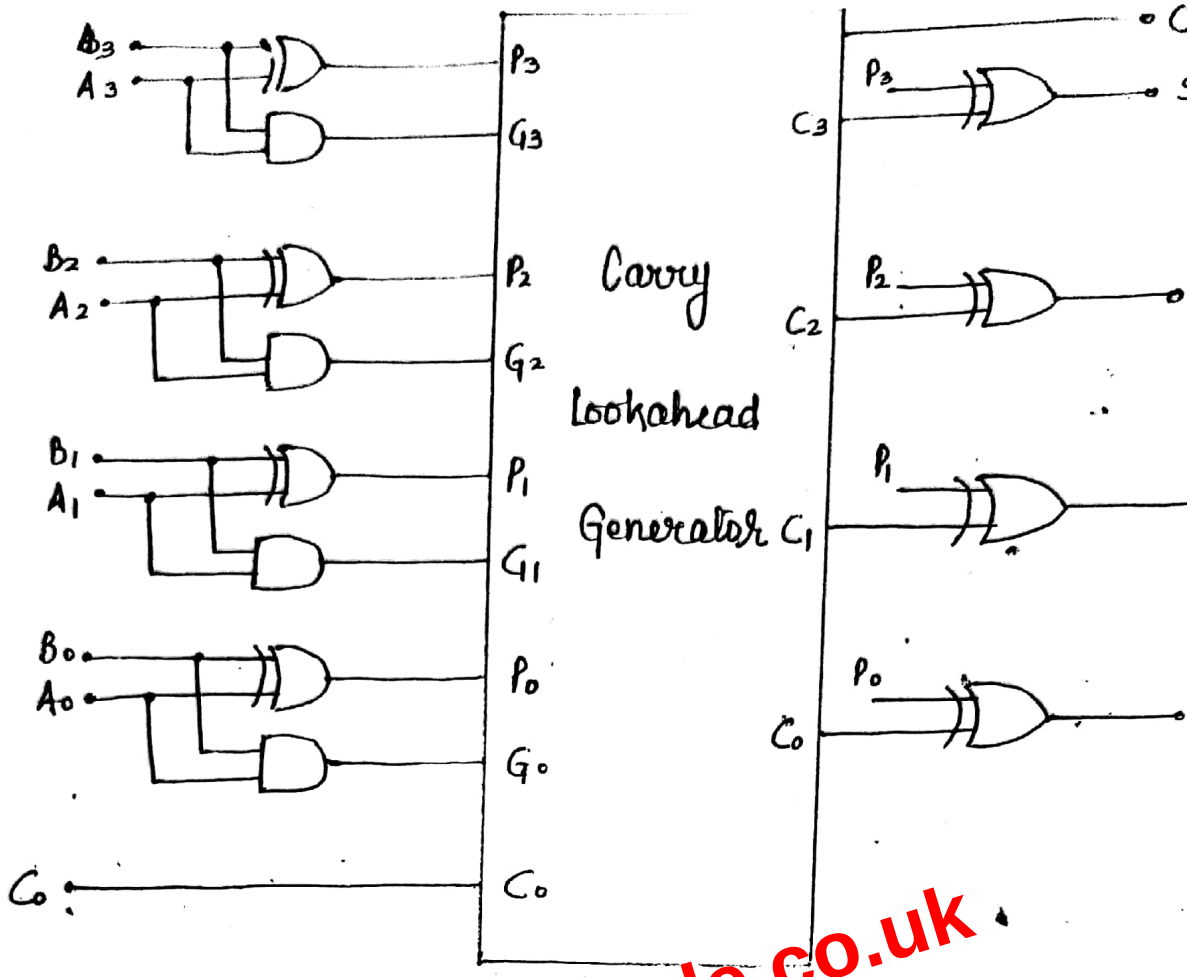
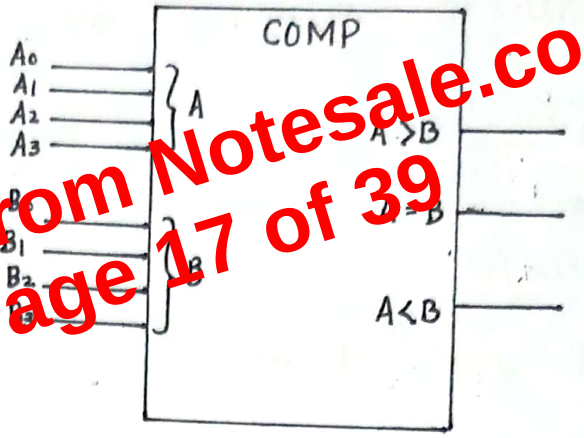


fig : 4-bit adder with carry lookahead.

logic symbol for a 4-bit comparator.



PRIORITY ENCODER

In the encoder defined above if D_3 & D_6 are 1 simultaneously, the output will be 111 because all three outputs are equal to 1. This (111) doesnot represent either binary 3 or 6.

The priority encoder establish an input priority to ensure that only one input is encoded. If higher priority is given to inputs with higher subscript nos, & if both D_3 & D_6 are 1 at the same time, the o/p will be 110 because D_6 has higher priority than D_3.

Another ambiguity in the octal to binary encoder is that an o/p with all 0's is generated when all the inputs are 0; this o/p is the same as when D_0 is equal to 1. This discripency is resolved by providing one more o/p to indicate that atleast one input is equal to 1.

INPUTS				OUTPUTS		
D_0	D_1	D_2	D_3	X	Y	V
0	0	0	0	X	0	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1

V → valid bit indicator. It is 1 when one or more inputs are equal to 1.
If all inputs are 0, there is no valid input & V = 0.

X → in the input column X represents either 1 or 0.
ie X100 represents the two minterms 0100 & 1100.

Priority level : $D_3 > D_2 > D_1 > D_0$

The maps for simplifying outputs x & y can be derived by replacing each X in a row first by 0 & then by 1.

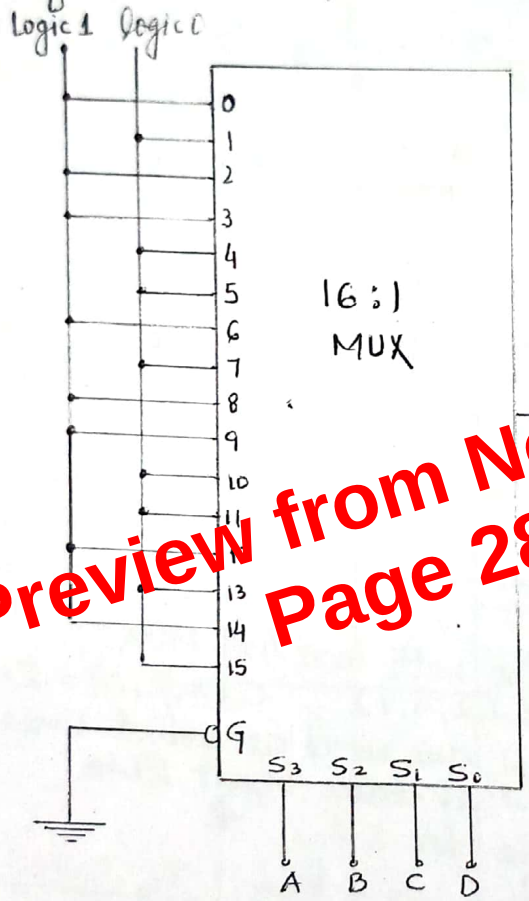
Logic Function Generation

For using the multiplexer as a logic element, either the TT or one of the std forms of logic expression must be available.

Ex: Implement the expression using a multiplexer

$$F(A,B,C,D) = \Sigma m(0,2,3,6,8,9,12,14)$$

Sol: Since there are 4 variables, therefore, a MUX with four select inputs is required.



G → Strobe or enable input which helps in cascading & it is generally active-low, which means it performs its intended operation when it is low.

Ex: Implement the previous function using 8:1 MUX.

Sol: $F(A,B,C,D) = \Sigma m(0,2,3,6,8,9,12,14)$

Ex: An 8:1 MUX has inputs C, B, A connected to selection lines S_2, S_1 and S_0 resp. The data inputs I_0 through I_7 are:

a) $I_1 = I_2 = 0$ $\quad I_3 = I_5 = I_7 = 1$

$I_0 = I_4 = D$ $\quad I_6 = \bar{D}$

Det. the Boolean function that the MUX implements.

Sol: Implementation table can be constructed from the given result:

		A B C →							
		I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
$\bar{D}$	0	0	2	4	(6)	8	(10)	(12)	(14)
D	1	(1)	3	5	(7)	(9)	(11)	13	(15)
		D	0	0	1	D	1	$\bar{D}$	1

Solved for ~~C B~~ A B C D instead of D C B A.

The SOP is obtd by the above table by writing the circled term as follows:-

$F(A, B, C, D) = \Sigma m(1, 6, 7, 9, 10, 11, 12, 14, 15)$

This is the function, implemented by the 8:1 MUX.

b) $I_0 = I_3 = I_5 = I_6 = I_7 = 0$

$I_1 = D$ $\quad I_2 = 1$

$I_4 = \bar{D}$

Here, the implementation table is shown below.

The variable chosen for application to the inputs is the MSB variable D.

		I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
$\bar{D}$	0	0	1	(2)	3	(4)	5	6	7
D	1	8	(9)	(10)	11	12	13	14	15
		0	D	1	0	$\bar{D}$	0	0	0

Solved for D C B ~~A~~ ok

Having completed the third row, relevant entries in the first and second rows are highlighted. The Boolean exp. can now be written as

$$Y = \Sigma m(2, 4, 9, 10)$$
$$= \bar{D}\bar{C}B\bar{A} + \bar{D}C\bar{B}\bar{A} + D\bar{C}\bar{B}A + D\bar{C}B\bar{A}$$
$$= \bar{C}B\bar{A} + \bar{D}C\bar{B}\bar{A} + D\bar{C}\bar{B}A.$$