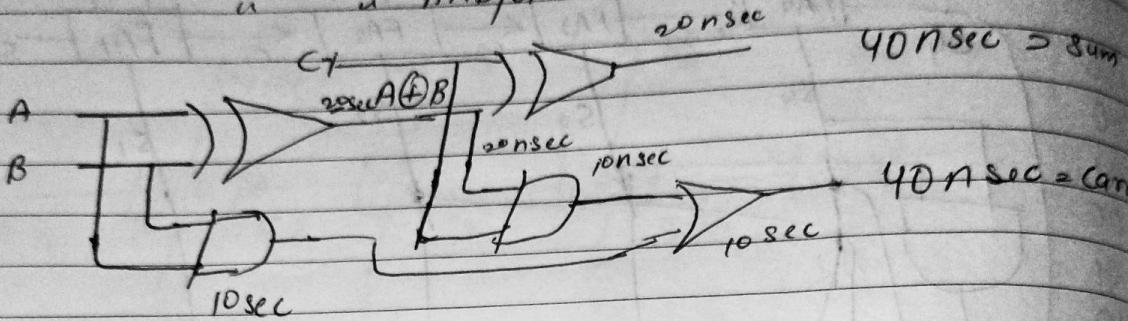


Look ahead carry

L-26

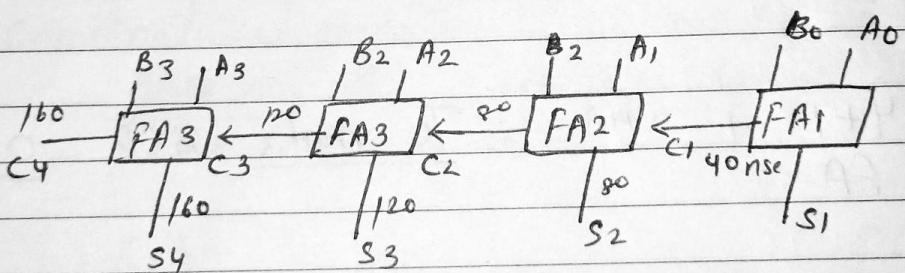
- \* Consider an implementation of F.A using 2 MA

→ delay of XOR — 20 nsec  
 " " AND/OR — 10 nsec



Total delay 40 sec

- \* 4 bit parallel adder — Total delay?  
 — 1 FA  $\rightarrow 40 \text{ nsec}$



160 nsec

1 FA delay

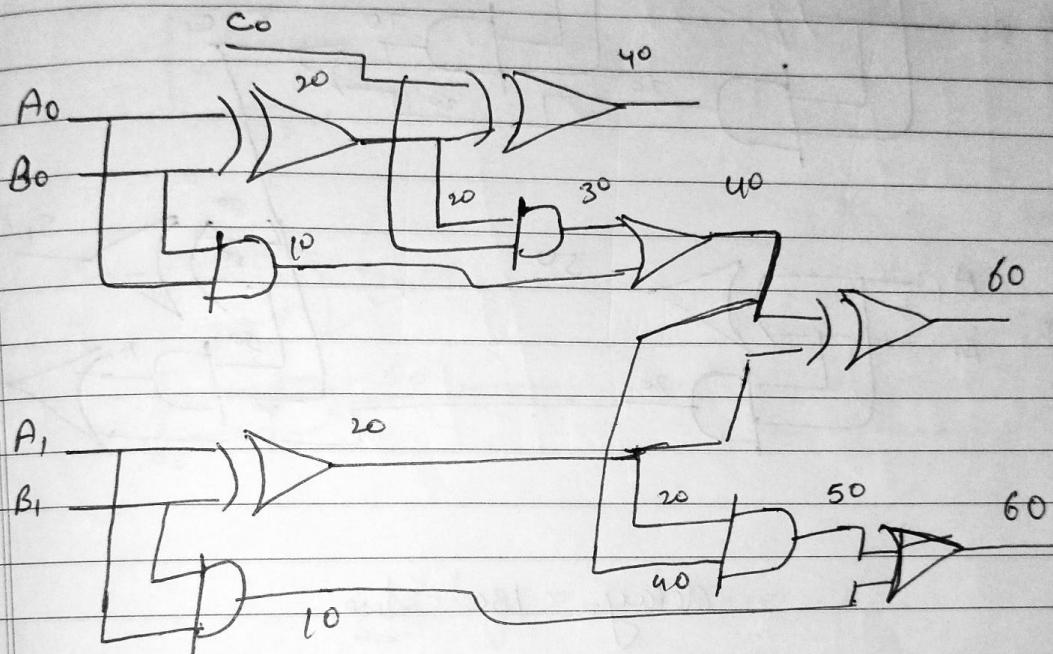
n bit parallel adder =  $m \times n$

2 bit parallel adder implemented using  
 & HA

XOR - 2 nsec

AND/OR - 10 nsec

$$\begin{matrix} & & & \text{FA} \\ n_1 & \xrightarrow{\quad} & \begin{bmatrix} e_1 & c_0 \\ A_1 & A_0 \\ B_1 & B_0 \\ c_2 & S_1 & S_0 \end{bmatrix} & \xrightarrow{\quad} \text{FA} \\ n_2 & \xrightarrow{\quad} & & = 2 \text{ FA} \Rightarrow 4 \text{ HA} \end{matrix}$$

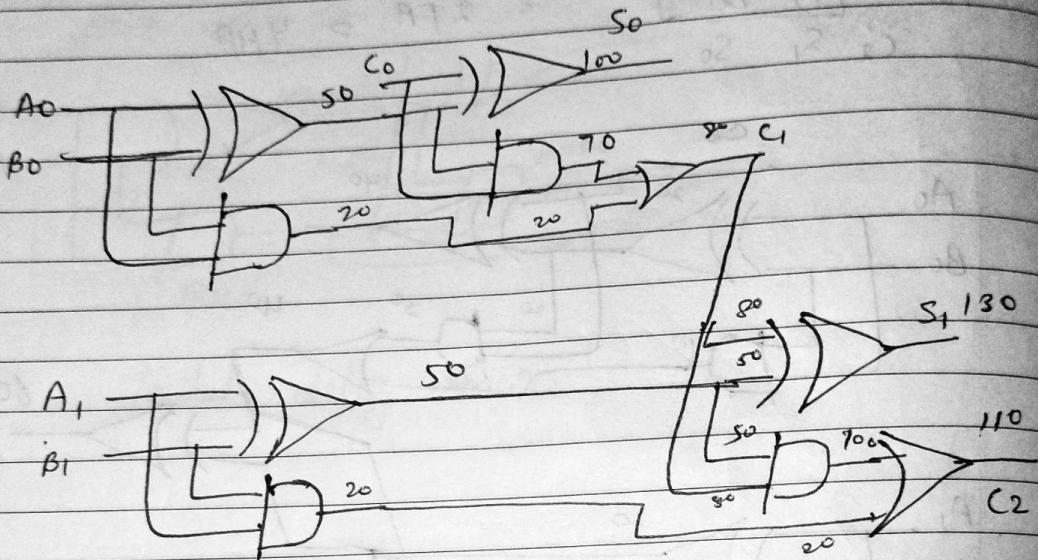


Total delay 60 nsec

$\alpha$ bit parallel adder  $\rightarrow$  delay?

HA

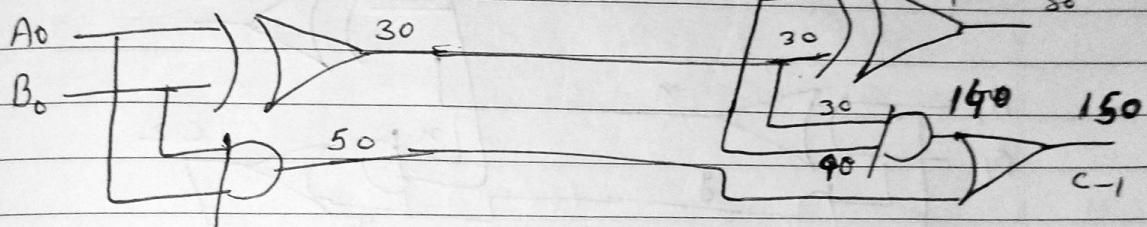
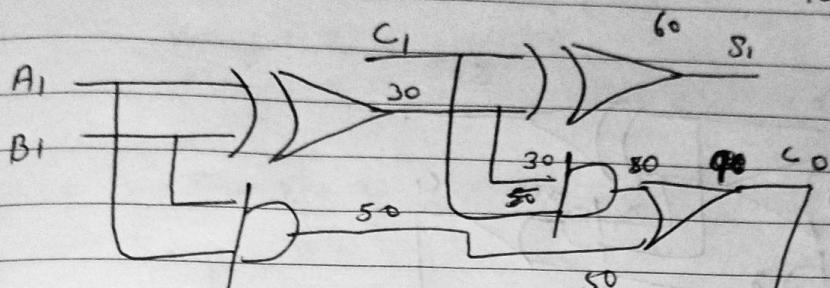
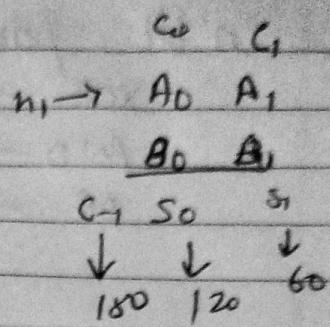
XOR  $\rightarrow$  50 nsec  
 AND  $\rightarrow$  20 "  
 OR  $\rightarrow$  10 "

 $c_1 \ c_0$  $n_1 \rightarrow A_1 \ A_0$  $n_2 \rightarrow \underline{B_1} \ B_0$  $(c_2 \ S_1 \ S_0)$ 

delay  $\approx 130$  nsec

2bit parallel adder

$$\begin{array}{l} \text{HA} \\ \text{XOR} - 30 \\ \text{AND} - 50 \\ \text{OR} - 10 \end{array}$$



$$\text{delay} = 150 \text{ ns}$$

~~n-bit parallel adder~~

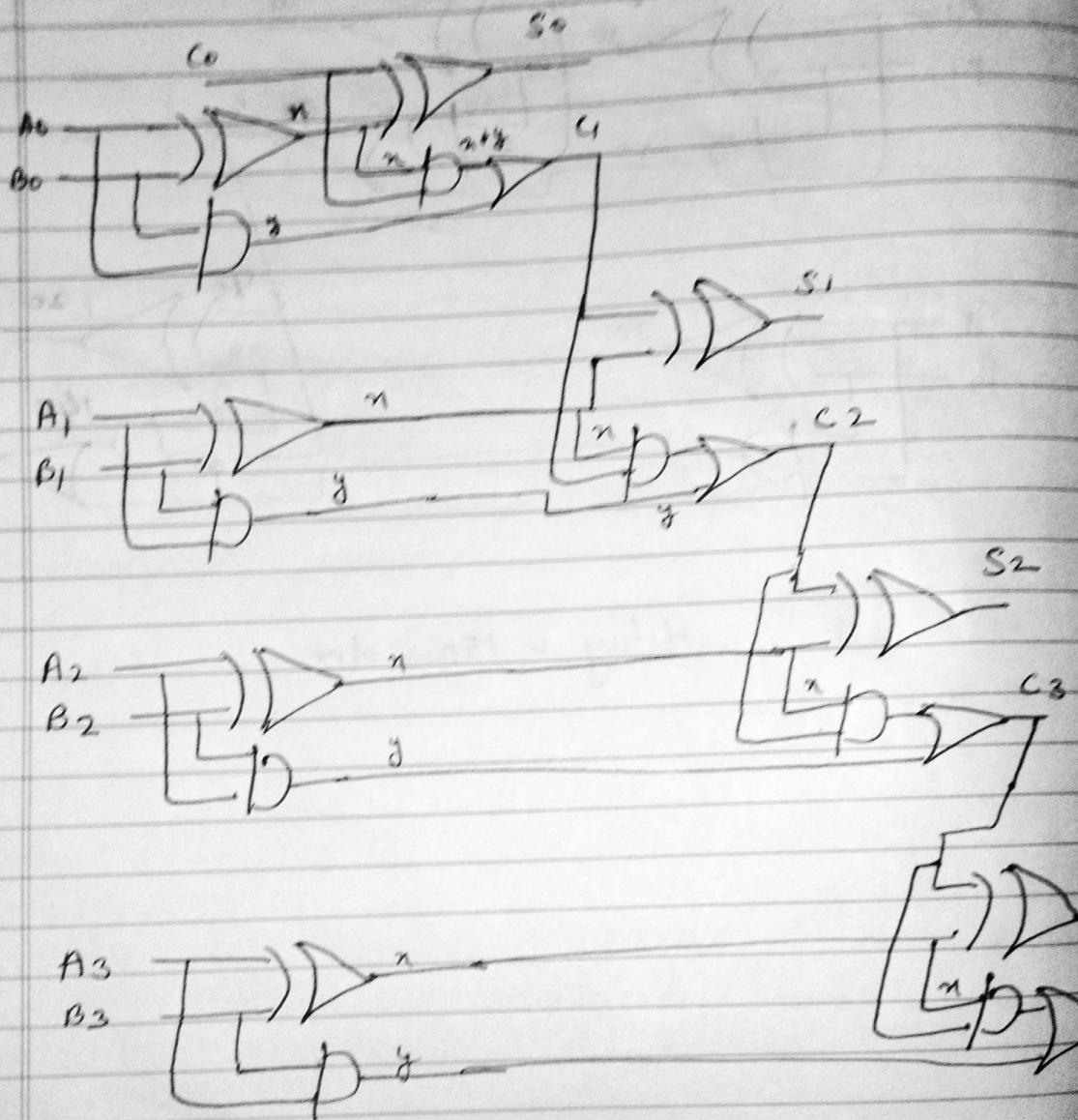
~~XOR = n only HA~~

~~AND = y~~

~~OR = z~~

✓ The Good Paper

$c_2$	$c_2$	$c_1$	$c_0$
$A_3$	$A_2$	$A_1$	$A_0$
$B_3$	$B_2$	$B_1$	$B_0$
$c_4$	$s_3$	$s_2$	$s_1$



$$\begin{aligned}
 S_0 &= 2n \\
 C_1 &= n+y+z \\
 S_1 &= n+y+z+n \\
 C_2 &= n+y+z+y+z \\
 S_2 &= n+y+z+y+z+n \\
 C_3 &= n+y+z+y+z+y+z \\
 S_3 &\rightarrow n+y+z+y+z+y+z+n \\
 C_4 &\rightarrow n+y+z+y+z+y+z+y+z
 \end{aligned}$$

$y+z$

$\frac{\text{Ans}}{2}$

$$\begin{aligned}
 S_0 &= 2n \\
 C_1 &= n+y+z
 \end{aligned}$$

$$\begin{aligned}
 \text{Total delay} &= S = 2n(n-1)^*(y+z) \\
 C &= n+y+z(n-1)^*(y+z)
 \end{aligned}$$

Formula for n bit parallel adder  
only using H.o.A

Total delay of n bit parallel adder if it is implemented using H.o.A only  
delay of XOR -  $n$ , AND -  $y$  and OR -  $z$

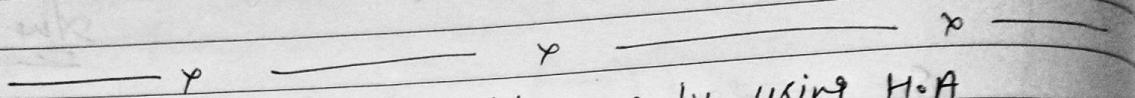
$$\begin{aligned}
 \text{Sum} &= 2n(n-1)(y+z) \\
 \text{Carry} &= n+y+z(n-1)(y+z)
 \end{aligned}$$

$$\text{Sum} = 2n + (n-1)(y+z) \Rightarrow n + \boxed{n + (n-1)(y+z)}$$

$$\text{Carry} = n+y+z(n-1)(y+z) \Rightarrow n + \boxed{(y+z)(n-1)(y+z)} \text{ diff}$$

if  $n > y+z$ , sum will decide  
the total delay

else if  $y+z > n$ , carry will decide  
the total delay  
else "any of them"



4bit parallel adder only using H.A  
 $\text{XOR} - 50, \text{AND} - 30, \text{OR} = 20$

$$\begin{aligned}\text{Sum} &= 2n + (n-1)(y+z) \\ &= 100 + 3 \times 50 \\ &= 250 \text{ nsec}\end{aligned}$$

$$\begin{aligned}\text{Carry} &= n+y+z+(n-1)(y+z) \\ &= 50+20+30+3 \times 50 \\ &= 250 \text{ nsec}\end{aligned}$$



# 8 bit parallel adder

Date: \_\_\_\_\_  
Page No.: \_\_\_\_\_

$\text{XOR} \rightarrow 20$ ,  $\text{AND} \rightarrow 50$ ,  $\text{OR} \rightarrow 10$

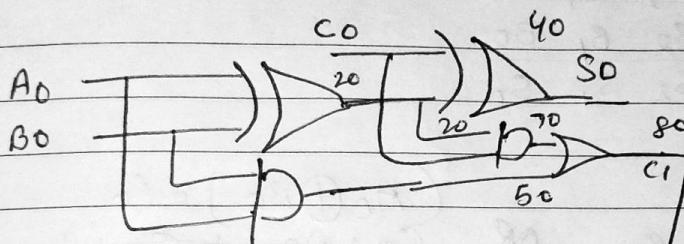
$$\begin{aligned}\text{Sum} &= 2^n (n-1) (y+z) \\ 40 &+ 7 \times 60 = 460\end{aligned}$$

$$\begin{aligned}\text{Carry} &= n+y+z (n-1)(y+z) \\ 80 &+ 7 \times 60 = 500 \quad \underline{\underline{\text{Ans}}}\end{aligned}$$

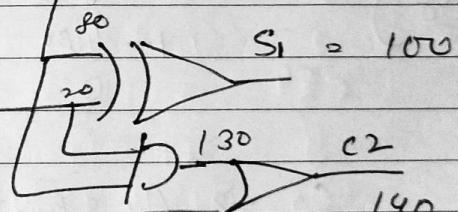
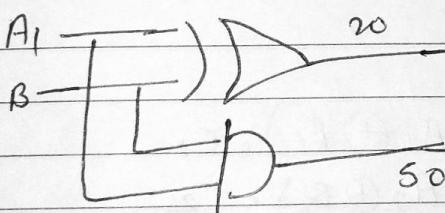
Total delay  $\rightarrow 500$

Same

ans without using formula  $c_1, c_0$



$$\begin{aligned}n_1 - A_1, A_0 \\ n_2 - B_1, B_0 \\ C_2, S_1, S_0\end{aligned}$$



$$\begin{aligned}s_0 &= 40 \\ c_1 &= 80\end{aligned}$$

$$\begin{aligned}s_1 &= 100 \\ c_2 &= 140\end{aligned}$$

$$\begin{aligned}s_0 \rightarrow 40 * 60 \times 7 = 460 \\ c \quad 80 * 60 \times 7 = 500\end{aligned}$$

~~Ans~~

disadvantage - 'n' bit parallel adder

as n increases then delay increase linearly  
if implemented by  $H \cdot A = O(n)$

actually starting Look ahead carry from here :-

### \* CLA generator

\* 4 bit Look ahead carry

$$\begin{array}{cccc} & c_3 & c_2 & c_1 & c_0 \\ n_1 \rightarrow & A_3 & A_2 & A_1 & A_0 \\ n_2 \rightarrow & B_3 & B_2 & B_1 & B_0 \\ c_4 & S_3 & S_2 & S_1 & S_0 \end{array}$$

possibilities }  $c_1 \Rightarrow A_0 \cdot B_0$  OR  $(A_0 \oplus B_0) \cdot C_0$

of carry }  $\begin{array}{c} 1 \\ | \\ 1 \\ 0 \end{array}$

$$c_2 \Rightarrow A_1 \cdot B_1 \text{ OR } (A_1 \oplus B_1) \cdot c_1$$

$$c_3 \Rightarrow A_2 \cdot B_2 \text{ OR } (A_2 \oplus B_2) \cdot c_2$$

$$c_4 \Rightarrow A_3 \cdot B_3 \text{ OR } (A_3 \oplus B_3) \cdot c_3$$

$\rightarrow$  carry generated at  $i^{\text{th}}$  stage

$$q_i = A_i \cdot B_i$$

$$p_i = A_i + B_i$$

$\rightarrow$  carry propagated at  $i^{\text{th}}$  stage

$$C_1 = q_0 + p_0 \cdot C_0$$

$$C_2 = q_1 + p_1 \cdot C_1 \Rightarrow q_1 + p_1 \cdot [q_0 + p_0 \cdot C_0] \Rightarrow q_1 + q_0 p_1 +$$

$$C_3 = q_2 + p_2 \cdot C_2$$

$$C_4 = q_3 + p_3 \cdot C_3$$

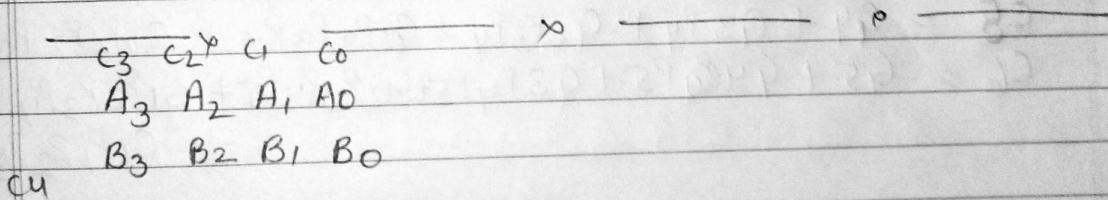
$$C_0 P_0 P_1$$

$$C_3 \Rightarrow q_2 + p_2 \cdot [q_1 + q_0 p_1 + C_0 P_0 P_1] \Rightarrow q_2 + q_1 P_2 +$$

$$+ q_0 P_1 P_2 + C_0 P_0 P_1 P_2$$

$$C_4 \Rightarrow q_3 + p_3 \cdot [q_2 + q_1 P_2 + q_0 P_1 P_2 + C_0 P_0 P_1 P_2] \Rightarrow$$

$$q_3 + q_2 P_3 + q_1 P_2 P_3 + q_0 P_1 P_2 P_3 + C_0 P_0 P_1 P_2 P_3$$



$$C_1 = q_0 + C_0 P_0$$

$$C_2 = q_1 + q_0 P_1 + C_0 P_0 P_1$$

$$C_3 = q_2 + q_1 P_2 + q_0 P_1 P_2 + C_0 P_0 P_1 P_2$$

$$C_4 = q_3 + q_2 P_3 + q_1 P_2 P_3 + q_0 P_1 P_2 P_3$$

$$G_i = A_i \cdot B_i$$

$$P_i = A_i \oplus B_i$$

$C_3$	$C_2$	$C_1 C_0$
$A_4$	$A_3$	$A_2 A_1$
$B_4$	$B_3$	$B_2 B_1$

$C_4$

$$C_1 = G_1 + C_0 P_1$$

$$C_2 = G_2 + G_1 P_2 + C_0 P_1 P_2$$

$$C_3 = G_3 + G_2 P_3 + G_1 P_2 P_3 + C_0 P_1 P_2 P_3$$

$$C_4 = G_4 + G_3 P_4 + G_2 P_3 P_4 + G_1 P_2 P_3 P_4 + C_0 P_1 P_2 P_3 P_4$$

$$G_i = A_i \cdot B_i$$

$$P_i = A_i \oplus B_i$$

$C_5$	$C_4$	$C_3$	$C_2$	$C_1$
$A_5$	$A_4$	$A_3$	$A_2$	$A_1$
$B_5$	$B_4$	$B_3$	$B_2$	$B_1$

$C_6$

$$C_2 = G_1 + C_1 P_1$$

$$C_3 = G_2 + G_1 P_2 + C_1 P_1 P_2$$
~~$$C_3 = G_3 + G_2 P_3 + G_1 P_2 P_3$$~~

$$C_4 = G_4 + G_3 P_4 + G_2 P_3 P_4 + G_1 P_2 P_3 P_4 + C_1 P_1 P_2 P_3 P_4$$

$$C_5 = G_5 + G_4 P_5 + G_3 P_4 P_5 + G_2 P_3 P_4 P_5 + G_1 P_2 P_3 P_4 P_5 + C_1 P_1 P_2 P_3 P_4 P_5$$

$$+ C_1 P_1 P_2 P_3 P_4 P_5$$

$C_3 \ C_2 \ C_1 \ C_0$

$A_3 \ A_2 \ A_1 \ A_0$   
 $B_3 \ B_2 \ B_1 \ B_0$

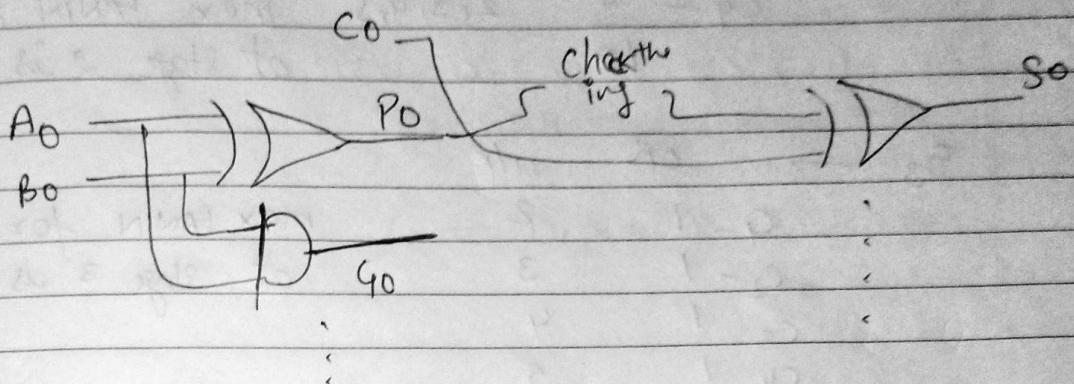
$c_4$

$$C_1 = G_0 + C_0 P_0$$

$$C_2 = G_1 + G_0 P_1 + C_0 P_0 P_1$$

$$C_3 = G_2 + G_1 P_2 + G_0 P_1 P_2 + C_0 P_0 P_1 P_2$$

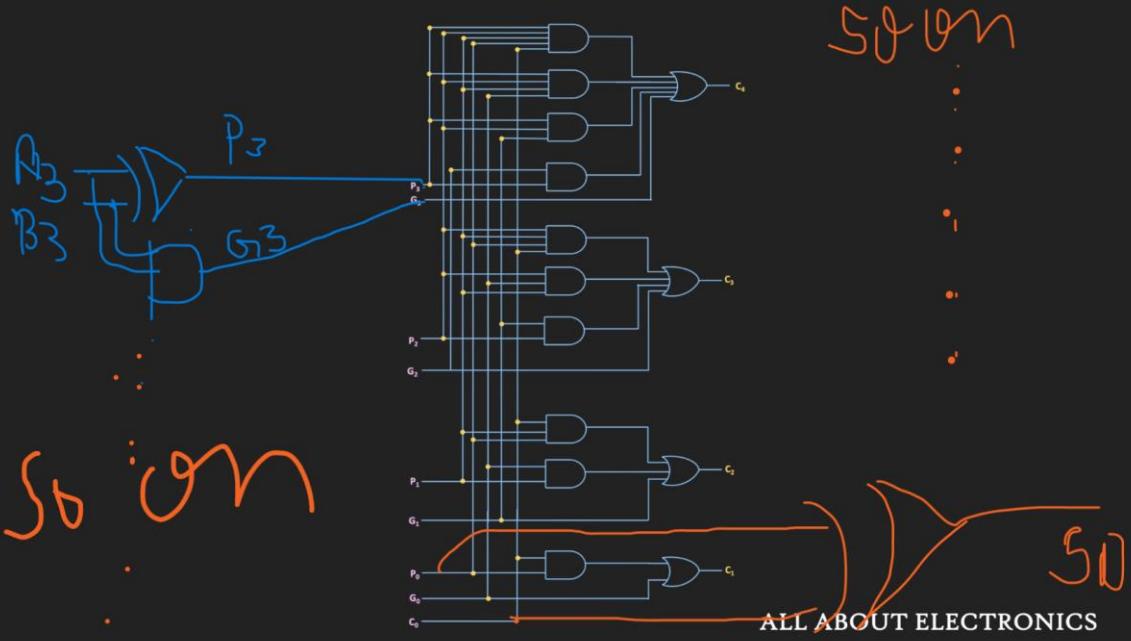
$$C_4 = G_3 + G_2 P_3 + G_1 P_2 P_3 + G_0 P_1 P_2 + C_0 P_0 P_1 P_2 P_3$$



use half adder to generate  $P_i$  and  $G_i$   
like above

will <sup>add</sup> ckt diagram of Look ahead carry generator

## Look Ahead Carry Adder logic circuit



Carry Look Ahead Adder (CLA) Explained

## 4-bit Carry Look Ahead Adder

$$C_1 = G_0 + P_0 C_0$$

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

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

$$C_4 = G_3 + P_3 C_3 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0$$

$$G_n = A_n \oplus B_n$$

$$P_n = A_n \oplus B_n$$