Datapath subsystems - Adder

Instructor – Dr. Madhav Rao

Half Adder

A	B	Sum	Carry-out Cout
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$S = A \oplus B$

$C_{out} = A \cdot B$

Full Adder

A	B	Cin	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$Sum = A \oplus B \oplus C_{in}$$

$$Sum = A\bar{B}\bar{C}_{in} + \bar{A}B\bar{C}_{in} + \bar{A}\bar{B}C_{in} + ABC_{in}$$

$$Cout = AB + BC_{in} + AC_{in}$$

$$= Majority(A, B, C_{in})$$

Full Adder

Propogate	A	B	Cin	S	Cout	Generate
0	0	0	0	0	0	0
	0	0	1	1	0	
1	0	1	0	1	0	0
	0	1	1	0	1	
1	1	0	0	1	0	0
	1	0	1	0	1	
0	1	1	0	0	1	1
	1	1	1	1	1	

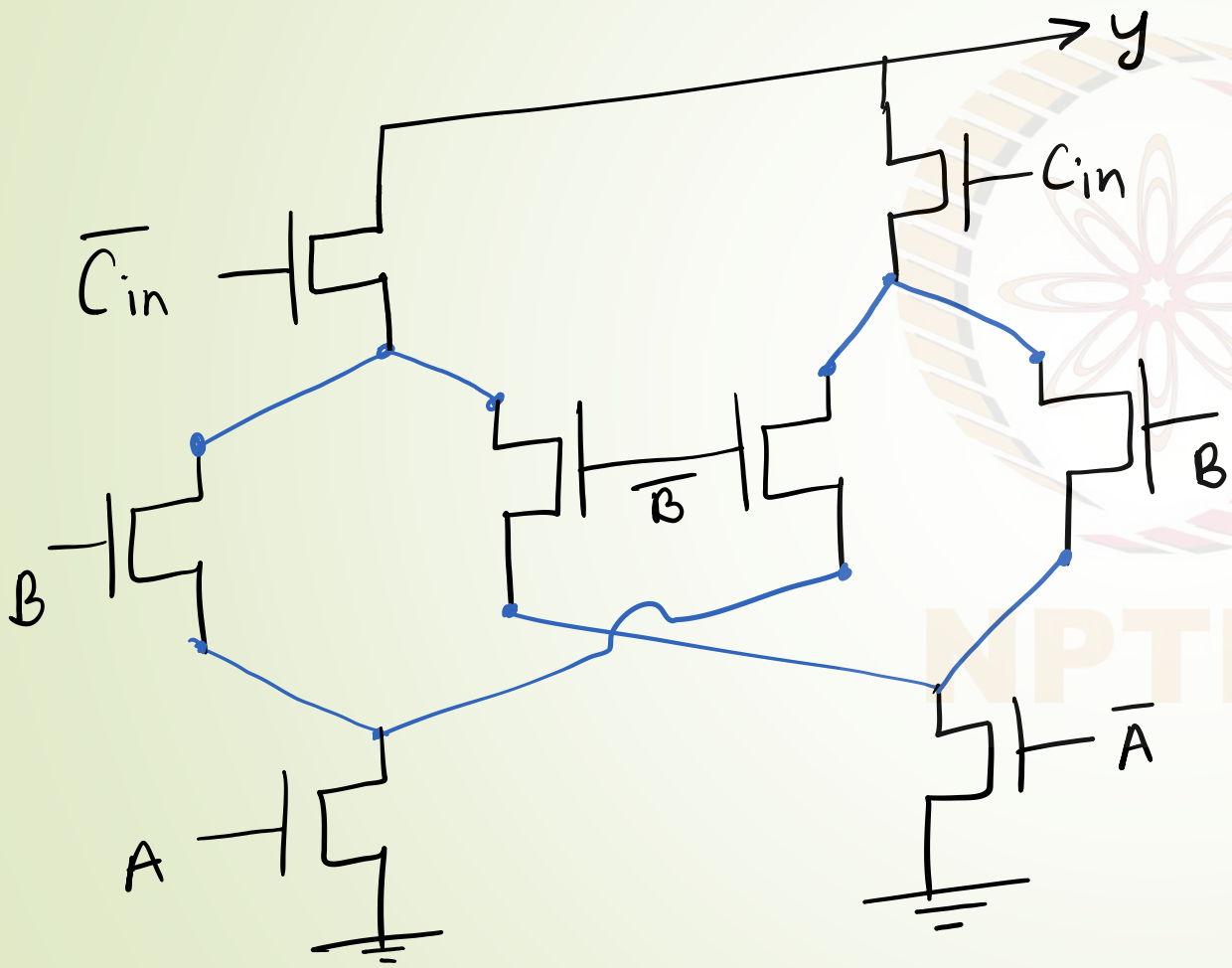
- Generate means without Cin , A and B should generate a Carry.
- Propagate means Cin should be propagated to Cout without generating a Carry.

Transistor level design

$$Sum = A\bar{B}\bar{C_{in}} + \bar{A}B\bar{C_{in}} + \bar{A}\bar{B}C_{in} + ABC_{in}$$

$$Sum = \overline{\bar{A}BC_{in} + A\bar{B}C_{in} + AB\bar{C_{in}} + \bar{A}\bar{B}\bar{C_{in}}}$$

$$Sum = \bar{A}B\bar{C}_{in} + A\bar{B}C_{in} + AB\bar{C}_{in} + \bar{A}\bar{B}\bar{C}_{in}$$

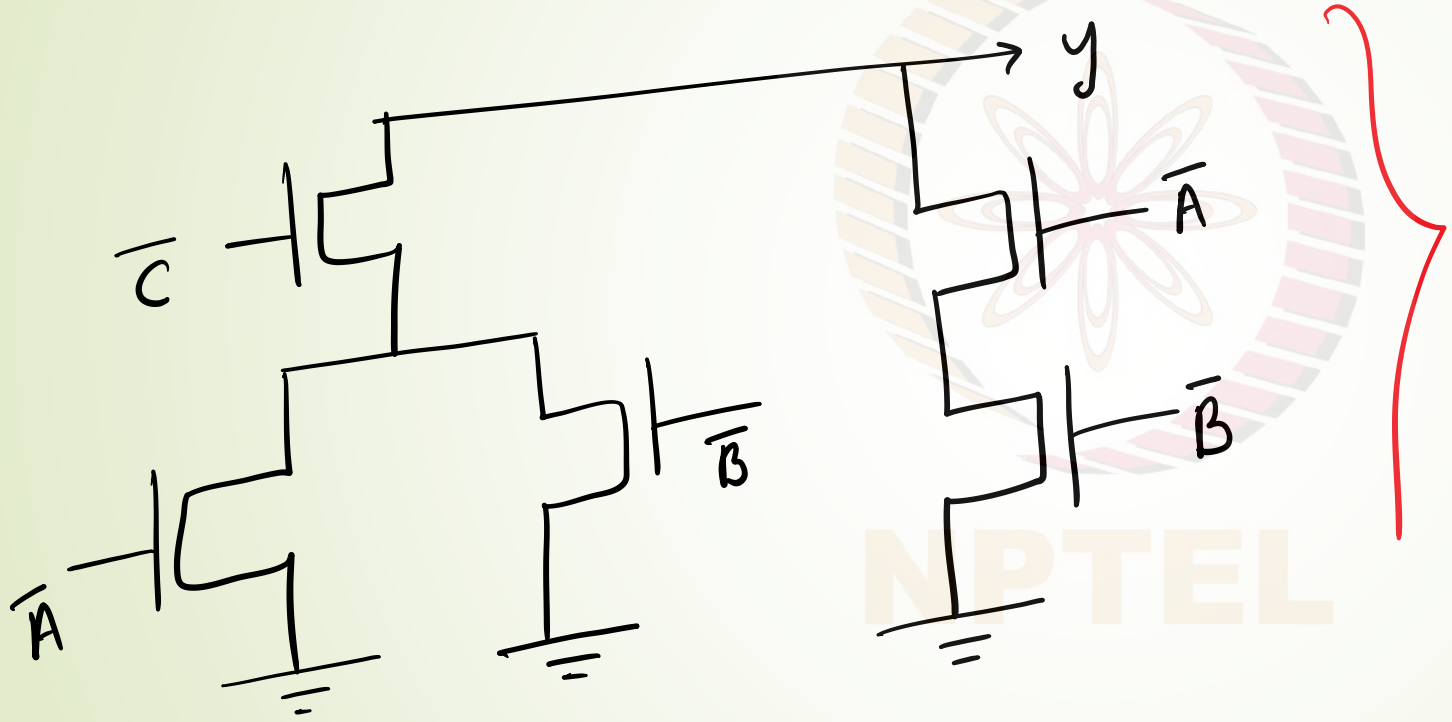


pull-down circuit

pull-up circuit → mirror topology

16 Transistors

$$C_{out} = \overline{\bar{A}\,\bar{B} + \bar{C}\,(\bar{A} + \bar{B})}$$



Pull-down

Pull-up : Mirror Topology

10 Transistors

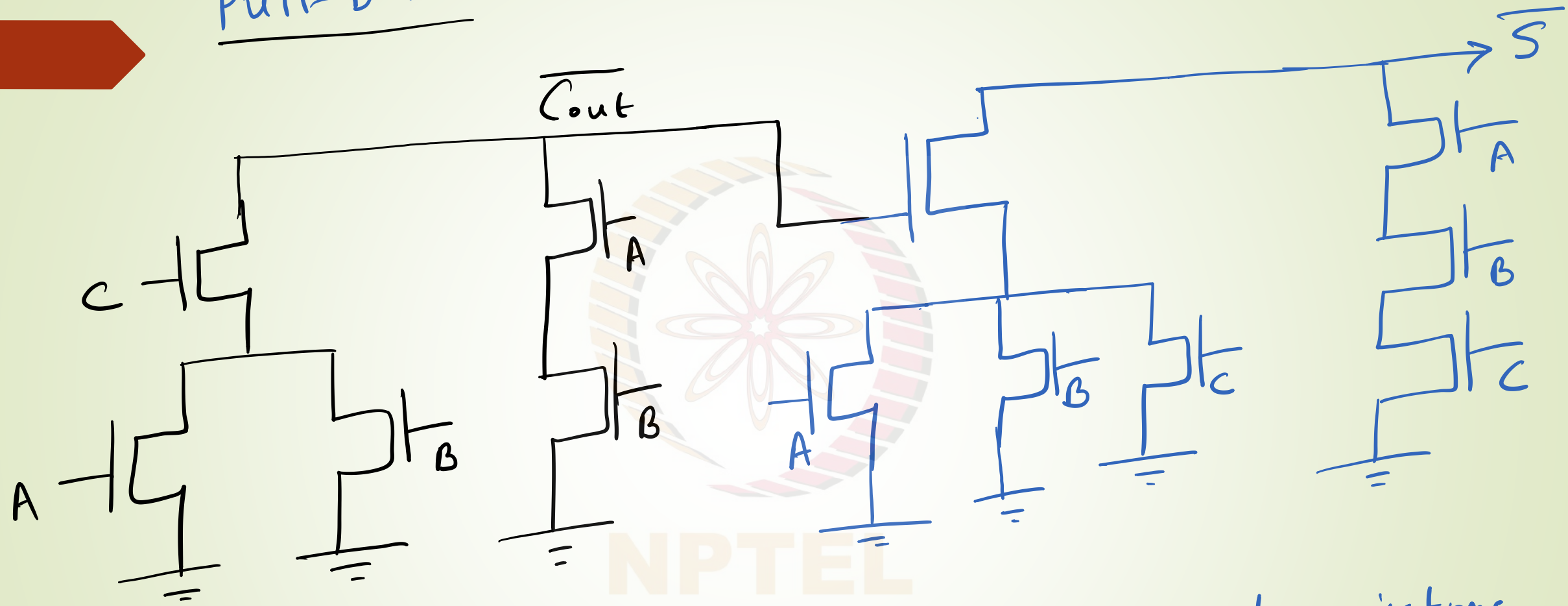
Sum & Cout & $\bar{A}, \bar{B}, \bar{C}$: $16 + 10 + 6 = 32$ Transistors

Compact Design

$$\overline{S} = \overline{ABC + (A+B+C)\,\overline{C_{out}}}$$

$$\overline{C_{out}} = \overline{AB + AC + BC} = \overline{AB + C(A+B)}$$

Pull-Down:

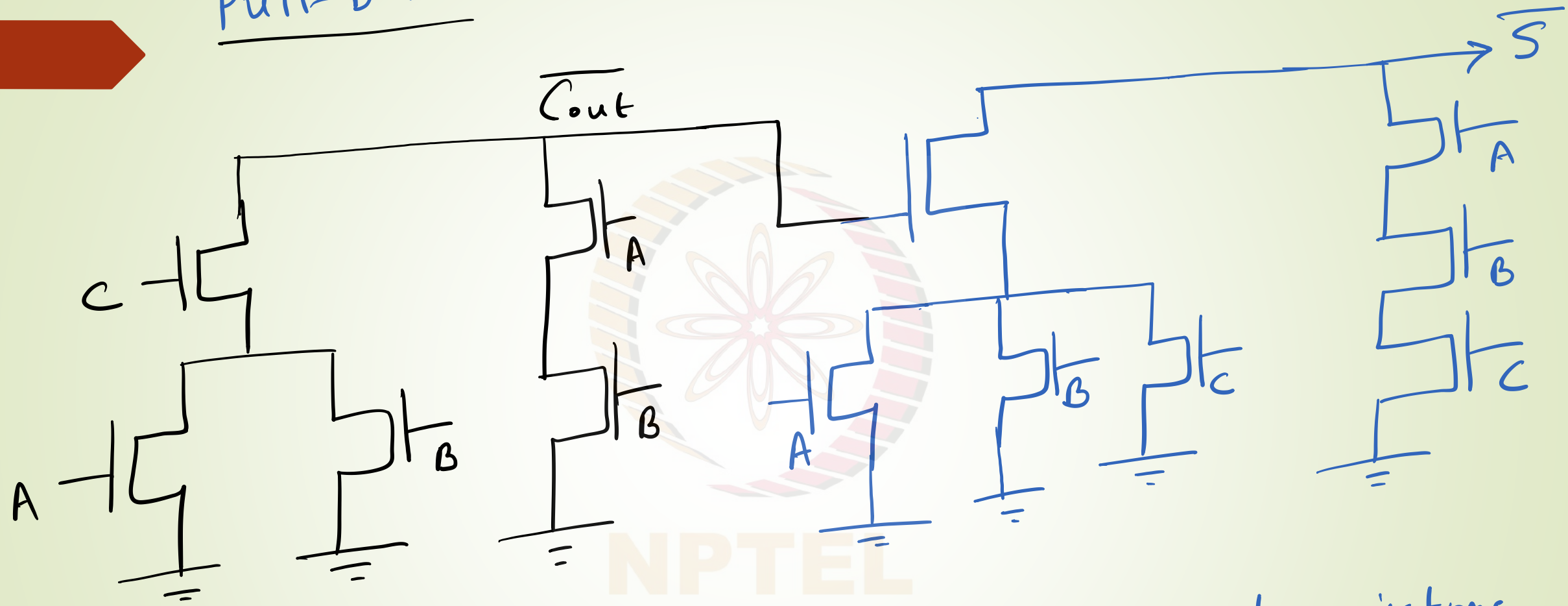


24 Transistors + 4 transistors = 28 transistors

Datapath subsystems - Adder

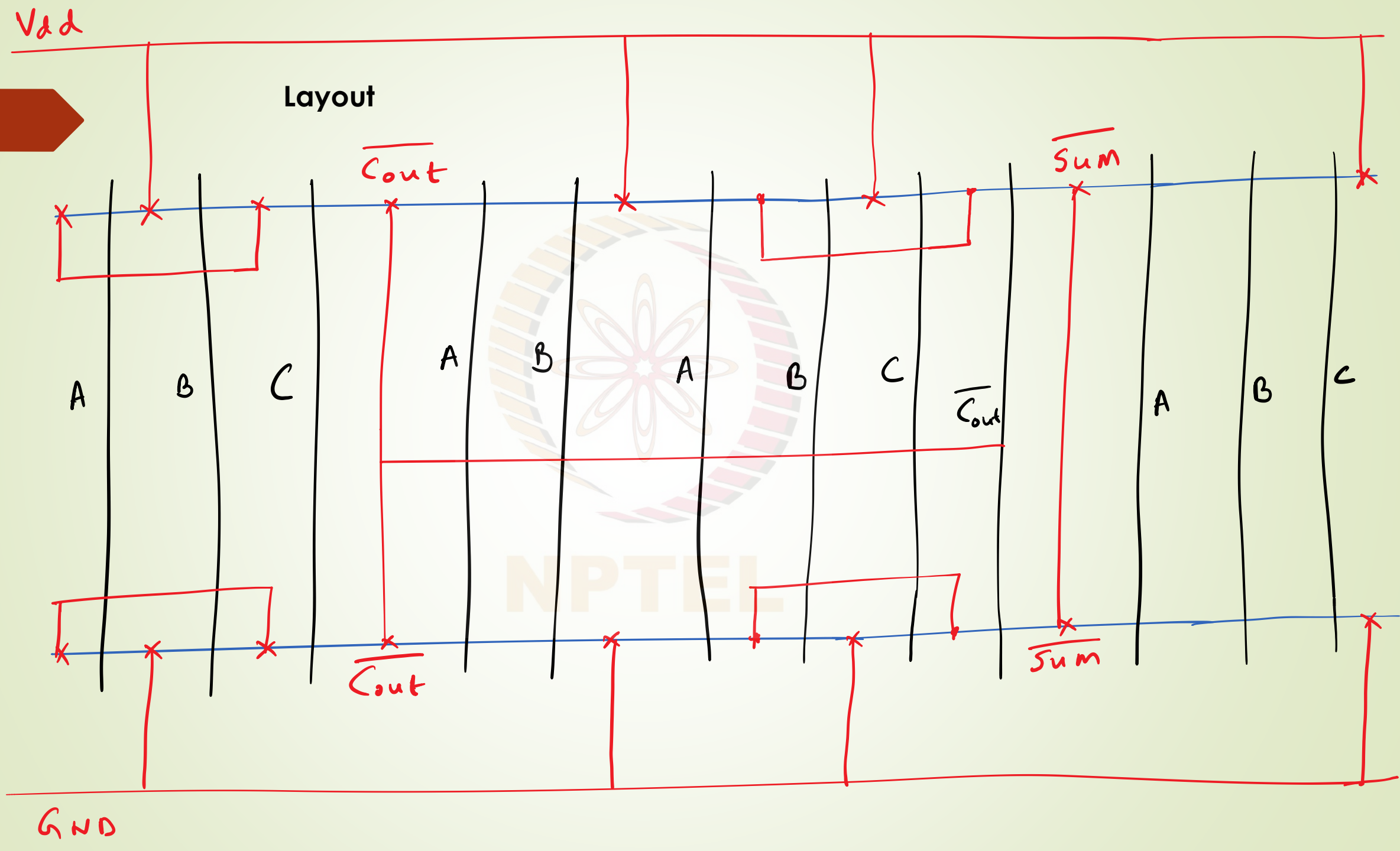
Instructor – Dr. Madhav Rao

Pull-Down:

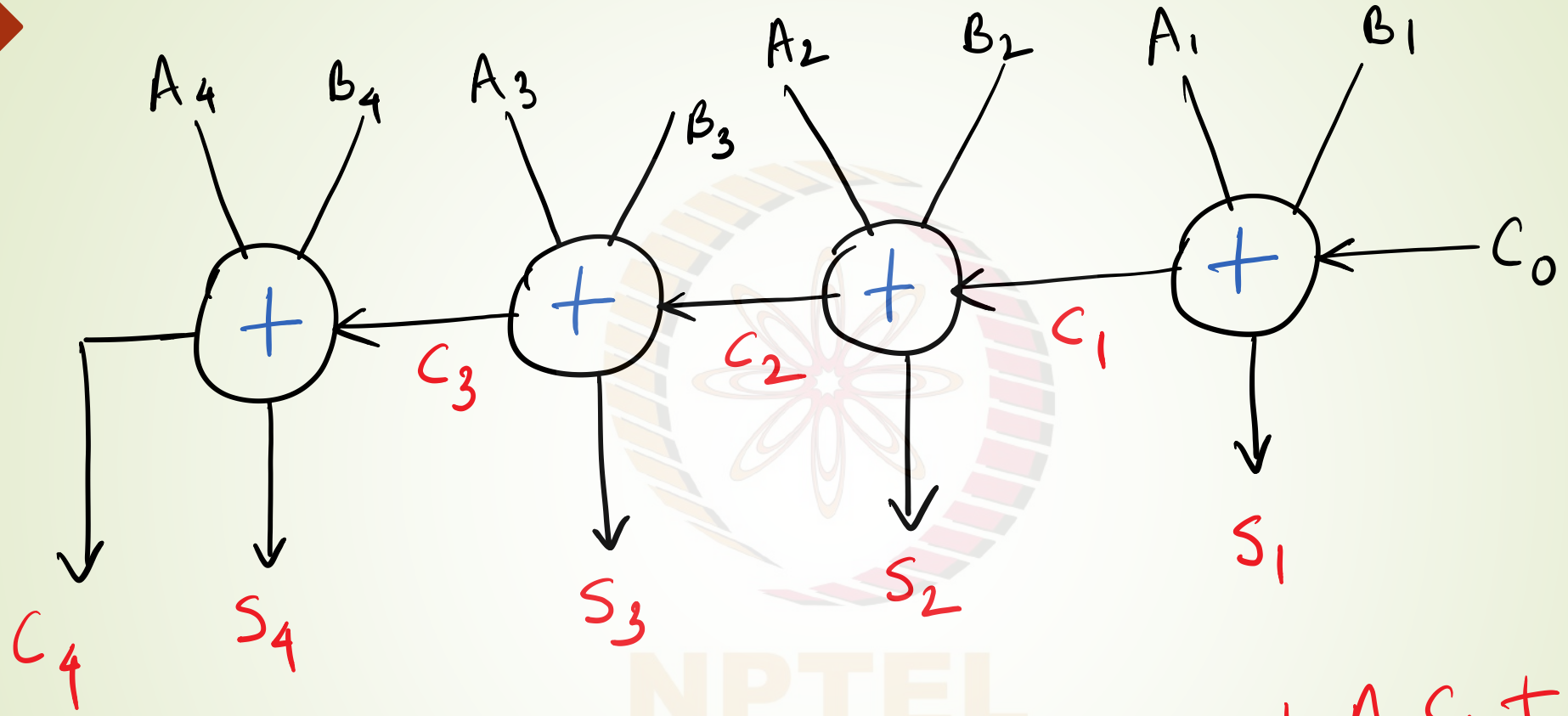


24 Transistors + 4 transistors = 28 transistors

Layout



Carry Ripple Adder



$S_1 = A_1 \oplus B_1 \oplus C_0$

$C_1 = A_1 B_1 + A_1 C_1 + B_1 C_0$

2-XOR

$S_1 = A_1 \oplus B_1 \oplus C_0$

$S_2 = A_2 \oplus B_2 \oplus C_1$

$S_{15} = A_{15} \oplus B_{15} \oplus C_{14}$

$S_{16} = A_{16} \oplus B_{16} \oplus C_{15}$

2-AND, 3-OR

$C_1 = A_1 B_1 + A_1 C_0 + B_1 C_0$

$C_2 = A_2 B_2 + A_2 C_1 + B_2 C_1$

$C_{15} = A_{15} B_{15} + A_{15} C_{14} + B_{15} C_{14}$

$C_{16} = A_{16} B_{16} + A_{16} C_{15} + B_{16} C_{15}$

$t_{C_0 \to S_{16}} = t_{XOR} + 15(t_{OR} + t_{AND})$

t_{OR} → $t_{3\text{-}OR}$, t_{AND} → $t_{2\text{-}AND}$

$t_{C_0 \to C_{16}} = 16(t_{OR} + t_{AND})$

t_{OR} → $t_{3\text{-}OR}$, t_{AND} → $t_{2\text{-}AND}$

Generate and Propagate Signals

Propogate	A	B	Cin	S	Cout	Generate
0	0	0	0	0	0	0
	0	0	1	1	0	
1	0	1	0	1	0	0
	0	1	1	0	1	
1	1	0	0	1	0	0
	1	0	1	0	1	
0	1	1	0	0	1	1
	1	1	1	1	1	

Generate (G) a carry if carry-out is truly independent of carry-in.

Propagate (P) a carry if its carry-out is true, when there is a carry-in.

$$G_i = A_i B_i$$

$$P_i = A_i \oplus B_i$$

Generate and Propagate bits

$$S = A \oplus B \oplus C_{in}$$

$$C_{out} = AB + BC_{in} + AC_{in}$$

$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

$$S_i = P_i \oplus C_{i-1}$$

$$G_{i-1:0} = C_{i-1}$$

Group generate signal

Definition of Group $G_{i:j}$ & $P_{i:j}$ still remains the same--- Generating Carry-out from the group irrespective of Carry-in to Group, and propagating a carry-out from the group, when there is a carry-in to the group.

$$G_{1:0} = G_{1:1} + P_{1:1} G_{0:0}$$

$$G_{1:0} = G_1 + P_1 G_0$$

$$= A_1 B_1 + (A_1 \overline{B_1} + \overline{A_1} B_1) C_0$$

$$= A_1 (B_1 + \overline{B_1} C_0) + \overline{A_1} B_1 C_0$$

$$= A_1 (B_1 + C_0) + \overline{A_1} B_1 C_0$$

$$G_{1:0} = A_1 B_1 + A_1 C_0 + \overline{A_1} B_1 C_0$$

$$= B_1 (A_1 + \overline{A_1} C_0) + A_1 C_0$$

$$G_{1:0} = B_1 A_1 + B_1 C_0 + A_1 C_0$$

$$G_{1:0} = C_1$$

$$\vdots$$

$$G_{2:0} = C_2$$

$$S_i = P_i \oplus G_{i-1:0}$$

$$G_{i:j} = G_{i:K} + P_{i:K}\, G_{K-1:j}$$

$P_{i:K}$ → group propogate signal

$$G_{N-1:0} = G_{N-1:K} + P_{N-1:K}\, G_{K-1:0}$$

$$S_N = P_N \oplus G_{N-1:0}$$

Datapath subsystems - Adder

Instructor – Dr. Madhav Rao

$$G_{i:j} = G_{i:K} + P_{i:K}\, G_{K-1:j}$$

$P_{i:K}$ → group propogate signal

$$G_{N-1:0} = G_{N-1:K} + P_{N-1:K}\, G_{K-1:0}$$

$$S_N = P_N \oplus G_{N-1:0}$$

$$G_{i:j} = G_{i:k} + P_{i:k} G_{k-1:j}$$

Group Generate signal will generate a carry-out if higher bits generate or lower bits generate-& higher bits propagate.

$$P_{i:j} = P_{i:k} P_{k-1:j}$$

Group Propagate signal will propagate a carry-out if higher and lower bits, both propagate.

#Example

$A = 1111$ $B = 0101$, $C_{in} = C_0 = 0$

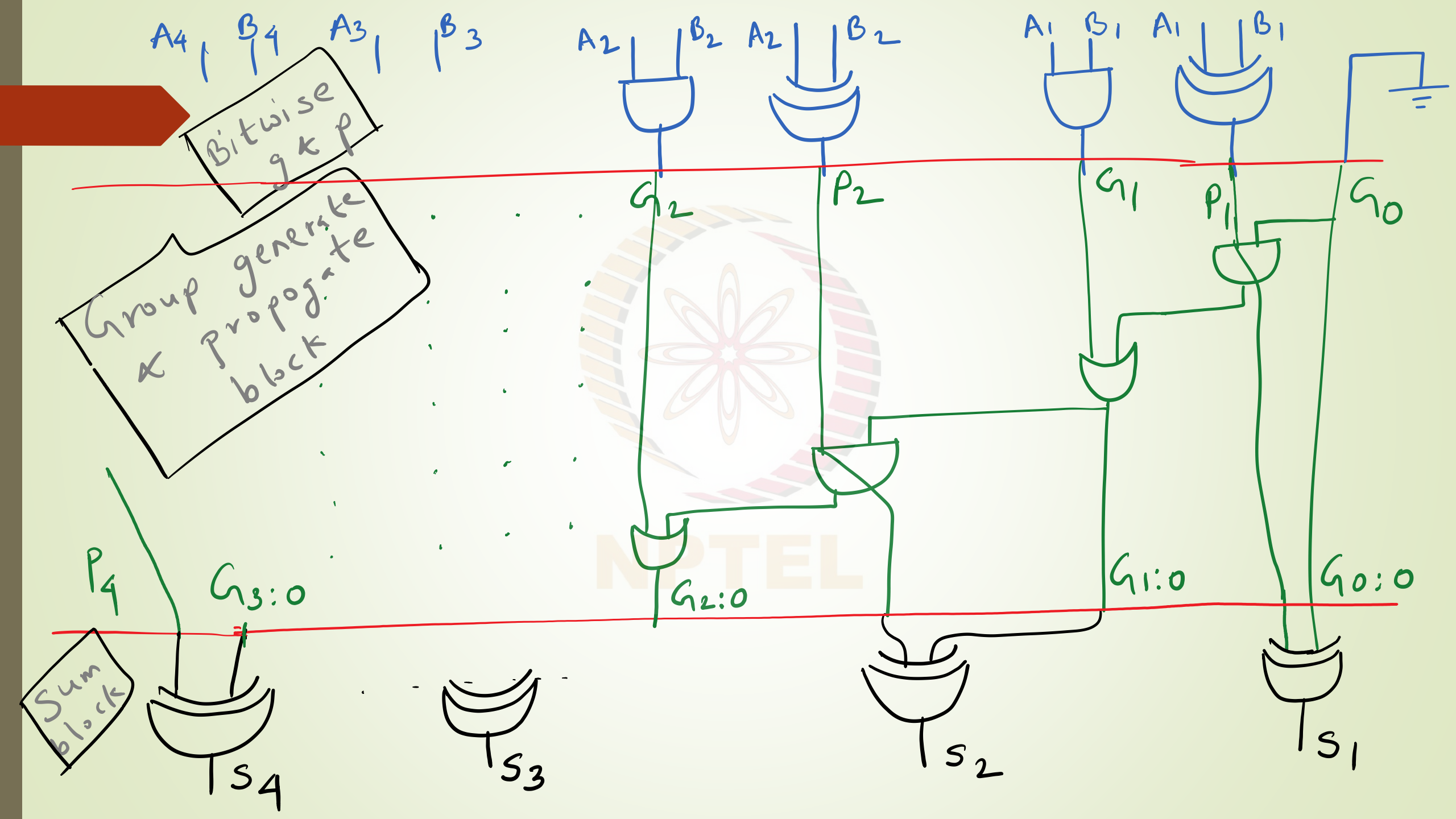
$P_i = A_i \oplus B_i$	$G_i = A_i B_i$	$G_{i:0} = G_{i:i} + G_{i-1:0} P_{i:i}$	$S_i = P_i \oplus G_{i-1:0}$
$P_1 = 0$	$G_1 = 1$	$G_{1:0} = G_1 + G_0 P_1$ $= 1$	$S_1 = 0 \oplus 0$ $= 0$
$P_2 = 1$	$G_2 = 0$	$G_{2:0} = G_2 + G_{1:0} P_2$ $= 1$	$S_2 = 1 \oplus 1$ $= 0$
$P_3 = 0$	$G_3 = 1$	$G_{3:0} = G_3 + G_{2:0} P_3$ $= 1$	$S_3 = 1 \oplus 0$ $= 1$
$P_4 = 1$	$G_4 = 0$	$G_{4:0} = G_4 + G_{3:0} P_4$ 1	$S_4 = 1 \oplus 1$ $= 0$
		$C_4 = 1$	$S = 0100$

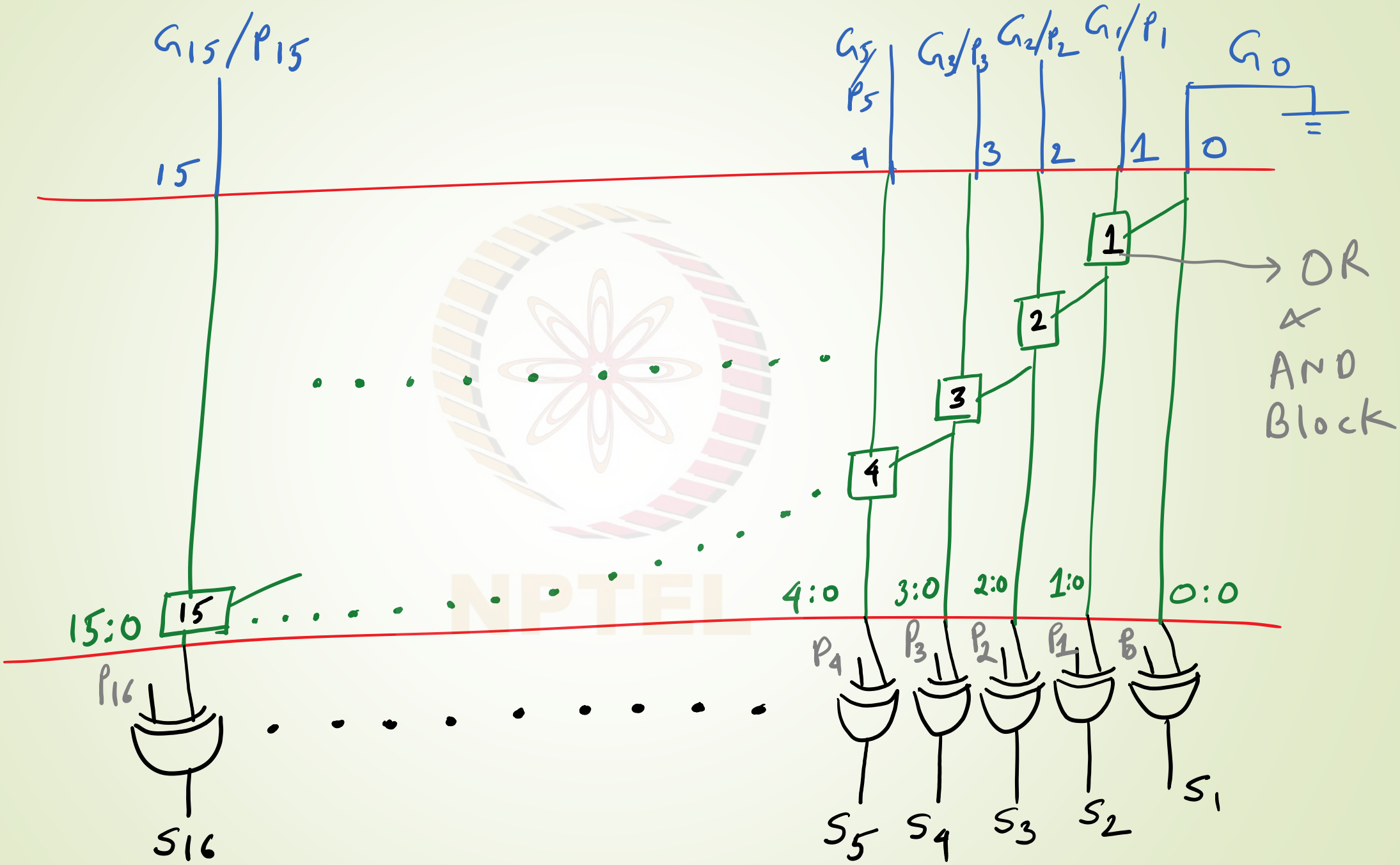
Carry Ripple Adder:

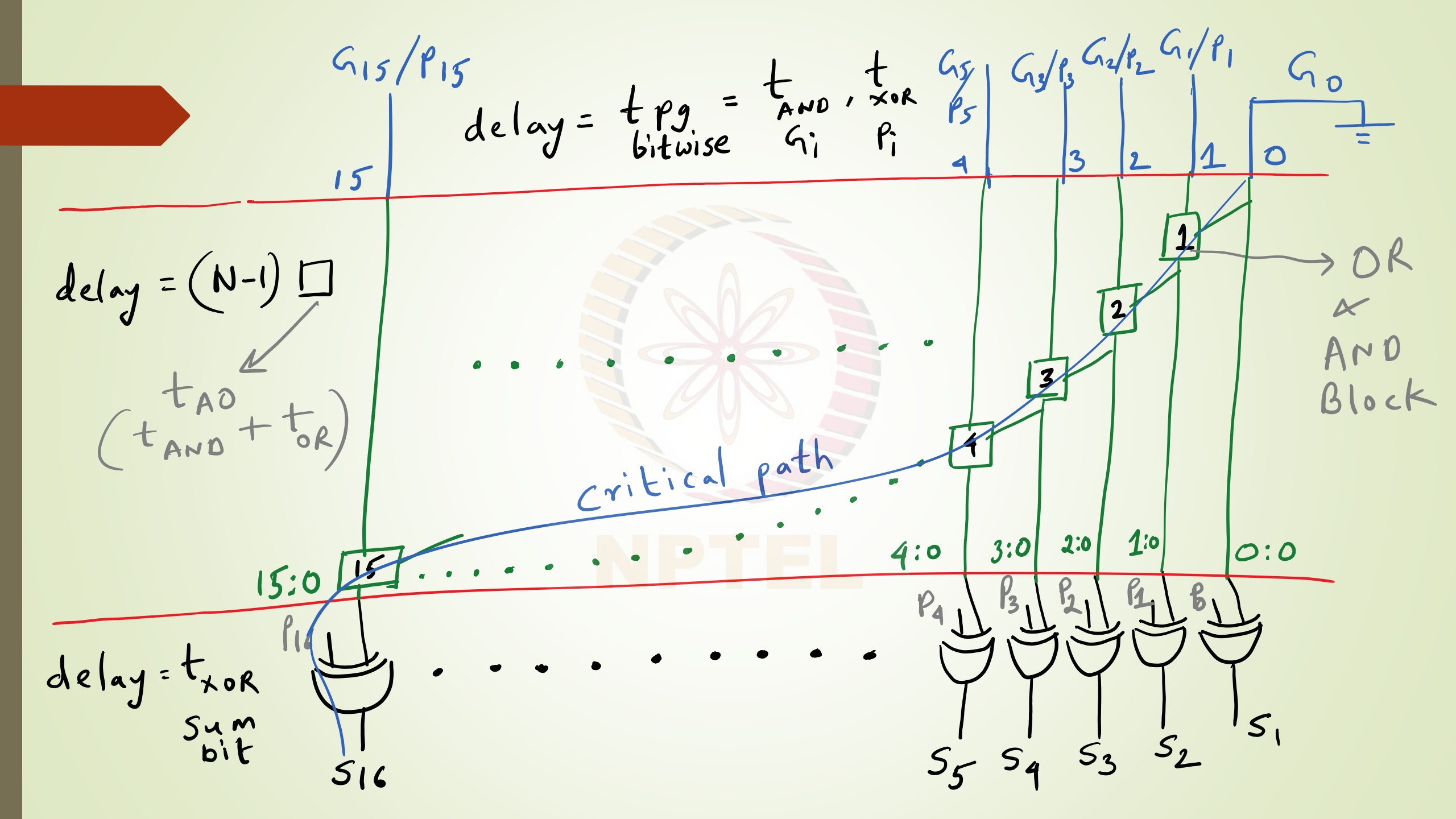
Generate G bits and Propagate P bits. $G_i = A_i B_i, \quad P_i = A_i \oplus B_i$

Generate Group $G_{i:0} = G_i + G_{i-1:0} P_i$

Sum bits $S_i = P_i \oplus G_{i-1:0}$


A4
B4
A3
B3
A2
B2
A2
B2
A1
B1
A1
B1
Bitwise g & p
Group generate & Propagate block
G2
P2
G1
P1
G0
P4
G3:0
G2:0
G1:0
G0:0
Sum block
S4
S3
S2
S1


G15/P15
15
G5/P5
4
G3/P3
3
G2/P2
2
G1/P1
1
G0
0
1
2
3
4
OR
&
AND
Block
15:0
15
4:0
3:0
2:0
1:0
0:0
P16
P4
P3
P2
P1
B
S16
S5
S4
S3
S2
S1


G_{15}/P_{15}
delay = t_{pg} = t_{AND}, t_{XOR}
bitwise G_i P_i
G_5 G_3/P_3 G_2/P_2 G_1/P_1 G_0
P_5
15
4 3 2 1 0
delay = (N-1) □
t_{AO}
(t_{AND} + t_{OR})
OR & AND Block
critical path
15:0
4:0 3:0 2:0 1:0 0:0
P_{15} P_4 P_3 P_2 P_1 B_0
delay = t_{XOR}
Sum bit
S_{15} S_5 S_4 S_3 S_2 S_1

P-G Architecture

$$t_{ripple\ Carry\ Adder} = t_{pg\ bitwise} + (N-1)\,\square + t_{2\text{-XOR}}$$

$t_{pg\ bitwise}$: t_{AND} (G_i), t_{XOR} (P_i)

$\square$: $t_{2\text{-AO}}$ — 2-AND, 2-OR gate

$t_{2\text{-XOR}}$: sum bit $P_i \oplus G_{i-1:0}$

Earlier -- Carry-Ripple architecture

$$t_{C_{in} \rightarrow S_N} = (N-1)\left(t_{3\text{-OR}} + t_{2\text{-AND}}\right) + t_{2\text{-XOR}}$$

P-G architecture is better

Datapath subsystems – Carry Skip Adder

Instructor – Dr. Madhav Rao

Carry Skip Adder (CSA) – Shortens the critical path by computing Group propagating signals

Carry-Ripple-Adder

$$G_{1:0} = G_{1:1} + P_{1:1} G_{0:0}$$

$$G_{2:0} = G_{2:2} + P_{2:2} G_{1:0}$$

Carry-skip-Adder

$$G_{4:1} = G_{4:4} + P_{4:4} G_{3:1}$$

$$G_{3:1} = G_{3:3} + P_{3:3} G_{2:1}$$

$$G_{2:1} = G_{2:2} + P_{2:2} G_{1:1}$$

$$\boxed{G_{4:0} = G_{4:1} + P_{4:1} G_{0:0}}$$

(I)

$$G_{4:1} = G_{4:4} + P_{4:4} G_{3:1}$$

$$G_{3:1} = G_{3:3} + P_{3:3} G_{2:1}$$

$$G_{2:1} = G_{2:2} + P_{2:2} G_{1:1}$$

(II)

$$G_{4:0} = G_{4:1} + P_{4:1} G_{0:0}$$

$$G_{8:0} = G_{8:5} + P_{8:5} G_{4:0}$$

$$G_{12:0} = G_{12:9} + P_{12:9} G_{8:0}$$

$$G_{16:0} = G_{16:13} + P_{16:13} G_{12:0}$$

(I)

$$G_{8:5} = G_{8:8} + P_{8:8} G_{7:5}$$

$$G_{7:5} = G_{7:7} + P_{7:7} G_{6:5}$$

$$G_{6:5} = G_{6:6} + P_{6:6} G_{5:5}$$

(I)

$$G_{12:9} = G_{12:12} + P_{12:12} G_{11:9}$$

$$G_{11:9} = G_{11:11} + P_{11:11} G_{10:9}$$

$$G_{10:9} = G_{10:10} + P_{10:10} G_{9:9}$$

(I)

$$G_{16:13} = G_{16:16} + P_{16:16} G_{15:13}$$

$$G_{15:13} = G_{15:15} + P_{15:15} G_{14:13}$$

$$G_{14:13} = G_{14:14} + P_{14:14} G_{13:13}$$

$$G_{4:1} = G_{4:4} + P_{4:4} G_{3:1}$$

$$G_{3:1} = G_{3:3} + P_{3:3} G_{2:1}$$

$$G_{2:1} = G_{2:2} + P_{2:2} G_{1:1}$$

If $G_{4:1} = 1$, implies either $G_{4:4}$ or $G_{3:3}$ or $G_{2:2}$ or $G_{1:1}$ has to be 1

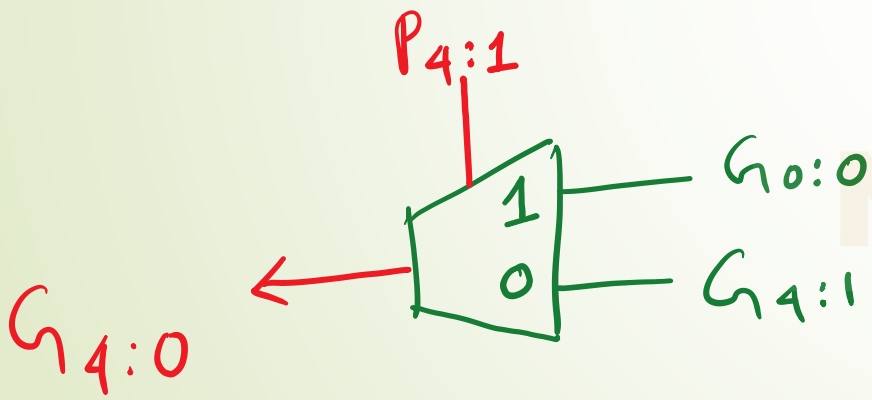
thereby $P_{4:1} = 0$

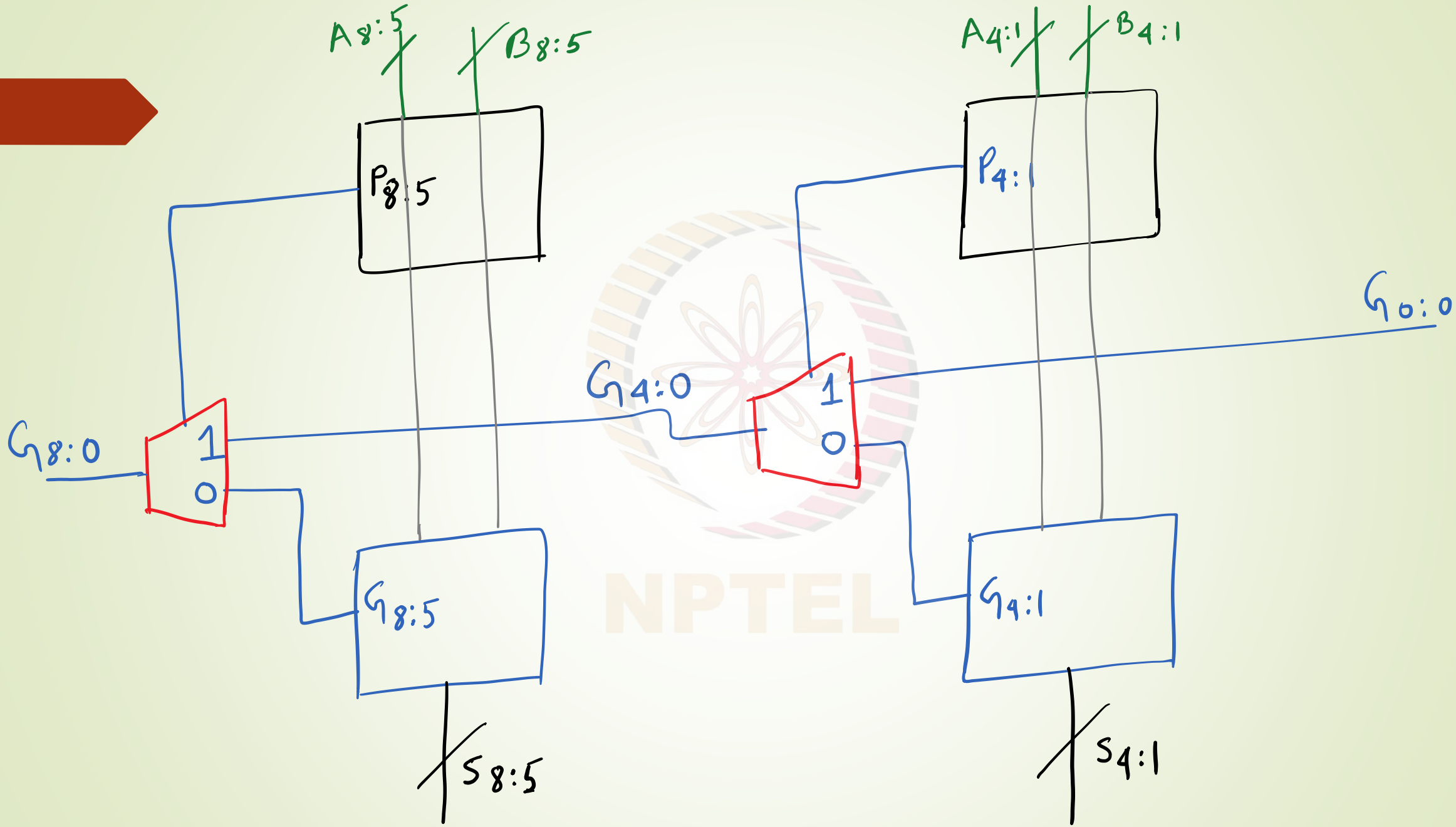
$$G_{4:0} = G_{4:1} + P_{4:1} G_{0:0}$$

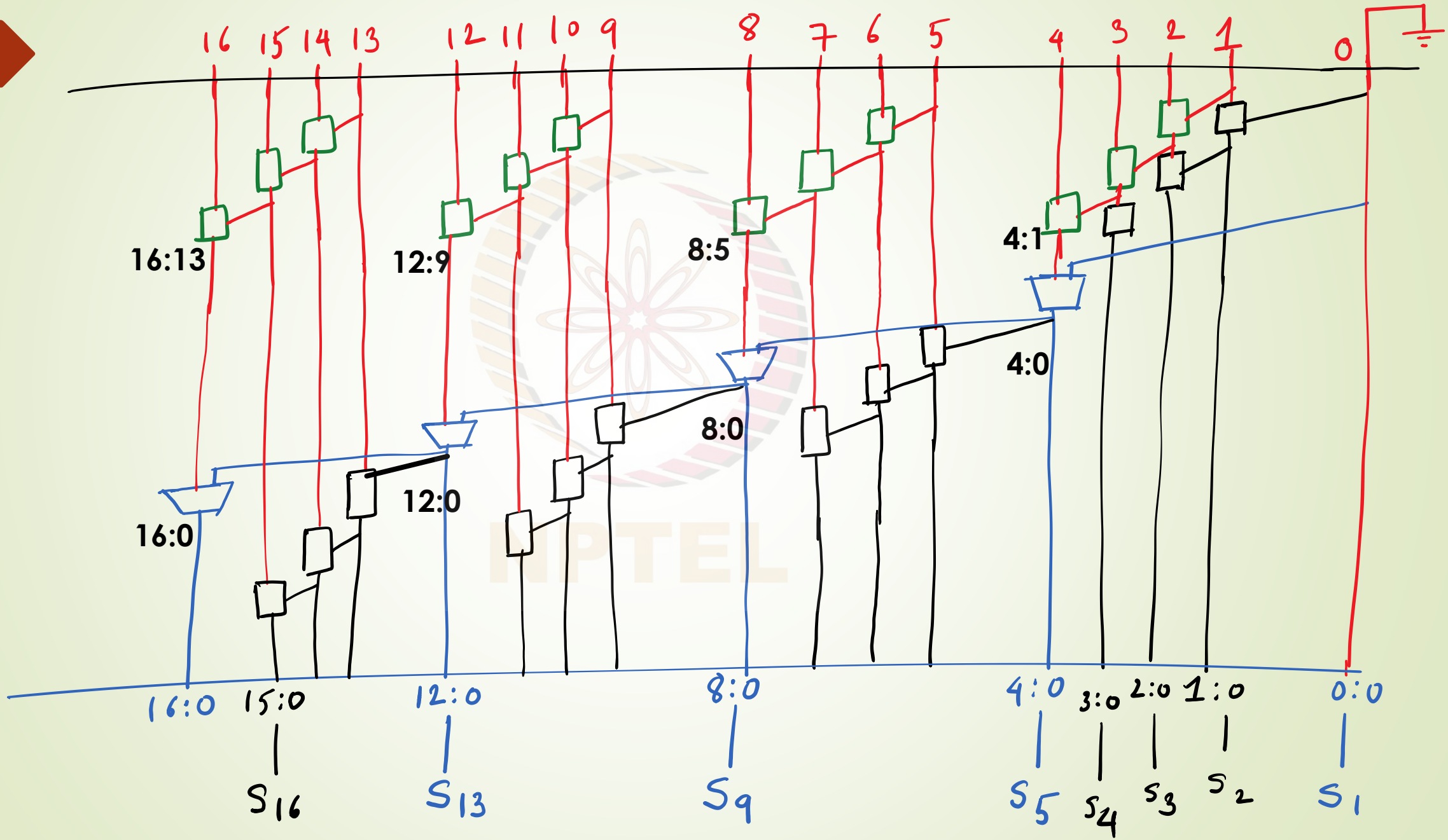
$$G_{4:0} = \overline{P_{4:1}}\, G_{4:1} + P_{4:1} G_{0:0}$$

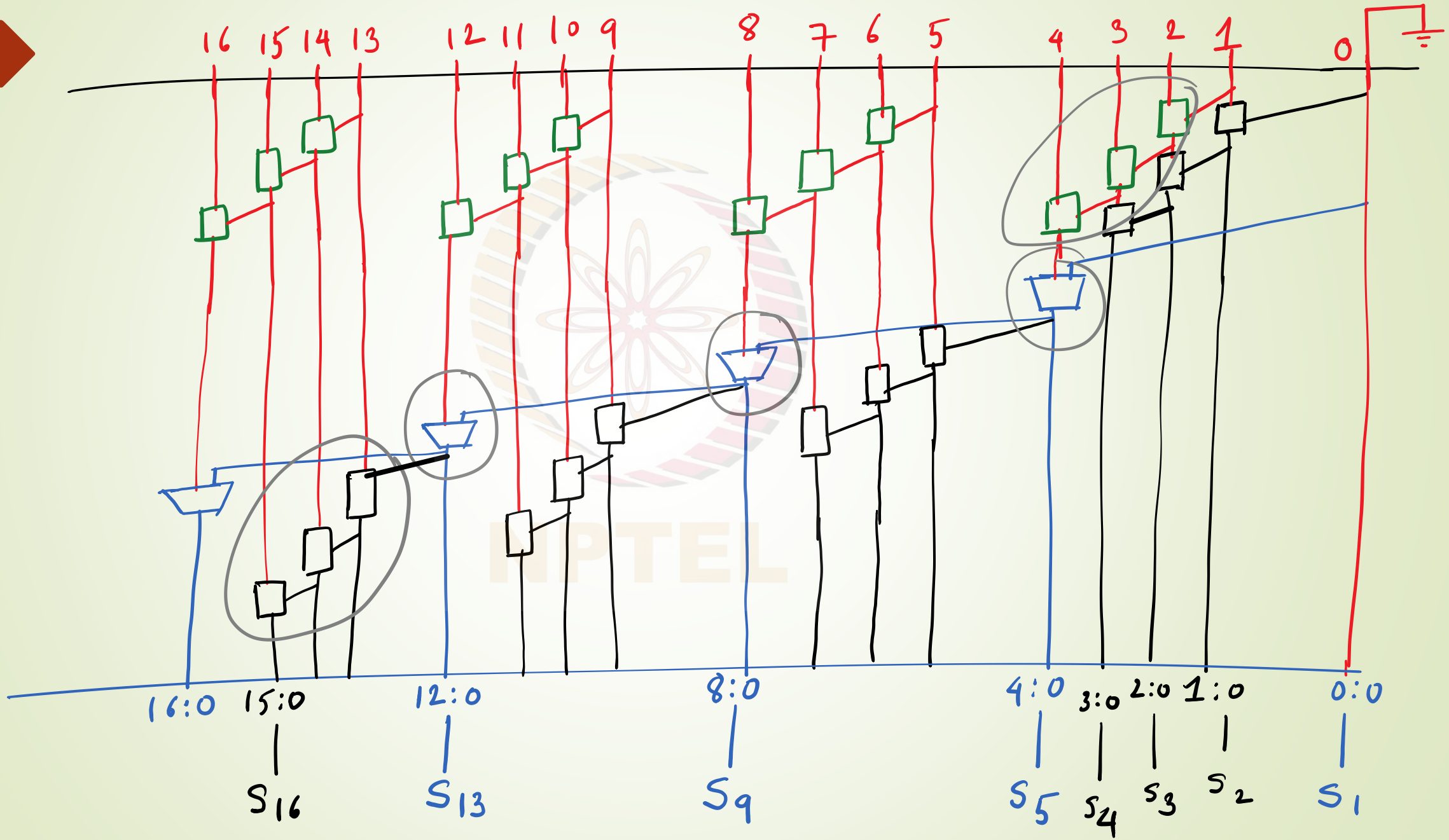
If $G_{4:1} = 0$, $P_{4:1} = 1$ or 0

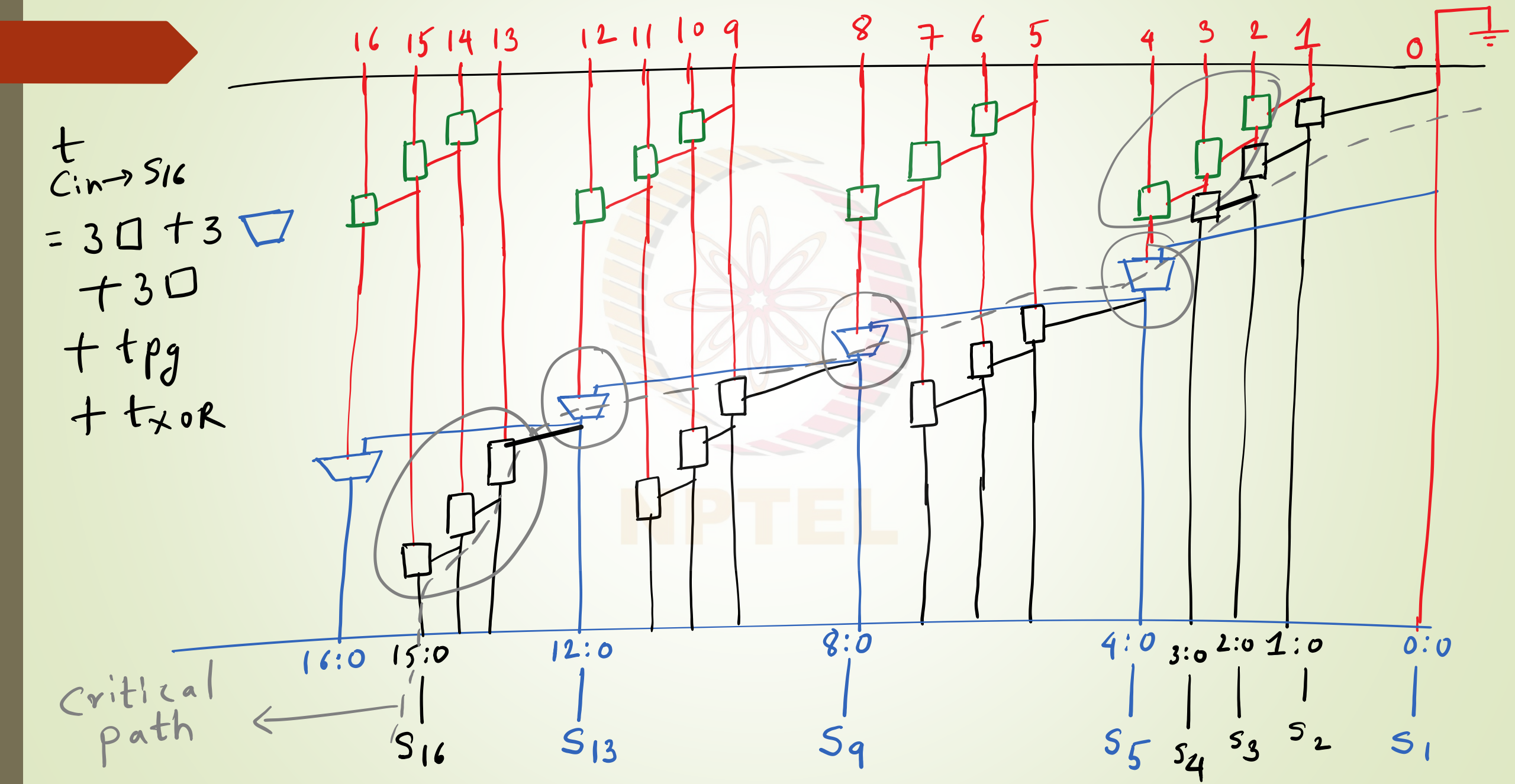
If $G_{4:1} = 1$, $P_{4:1} = 0$




A8:5
B8:5
P8:5
G8:0
1
0
G8:5
S8:5
G4:0
A4:1
B4:1
P4:1
1
0
G0:0
G4:1
S4:1


16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
16:13
12:9
8:5
4:1
16:0
12:0
8:0
4:0
16:0 15:0 12:0 8:0 4:0 3:0 2:0 1:0 0:0
S16 S13 S9 S5 S4 S3 S2 S1


16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
16:0 15:0 12:0 8:0 4:0 3:0 2:0 1:0 0:0
S16 S13 S9 S5 S4 S3 S2 S1


t
Cin→S16
= 3□ + 3
+ 3□
+ tpg
+ tXOR
16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
16:0 15:0 12:0 8:0 4:0 3:0 2:0 1:0 0:0
Critical path
S16 S13 S9 S5 S4 S3 S2 S1

$$t_{delay}\ (C_{in} \rightarrow S_{15}) = t_{pg\ bitwise} + 3\,t_{AO} + 3\,t_{mux} + 3\,t_{AO} + t_{XOR}$$

$3\,t_{AO}$: 1st group

$3\,t_{AO}$: last group

$$t_{delay} = t_{pg\ bitwise} + t_{XOR} + (k-1)\,t_{mux} + 2(N-1)\,t_{AO}$$

k groups & N bits in a group

Mux is slower than AND-OR gate

Datapath subsystems – Carry Look ahead Adder

Instructor – Dr. Madhav Rao

Carry Look ahead Adder (CLA) – Shortens the critical path by computing Group propagating signals

Carry-Ripple-Adder

$$G_{1:0} = G_{1:1} + P_{1:1} G_{0:0}$$

$$G_{2:0} = G_{2:2} + P_{2:2} G_{1:0}$$

Carry-Skip – Adder

Carry-Look-ahead-Adder

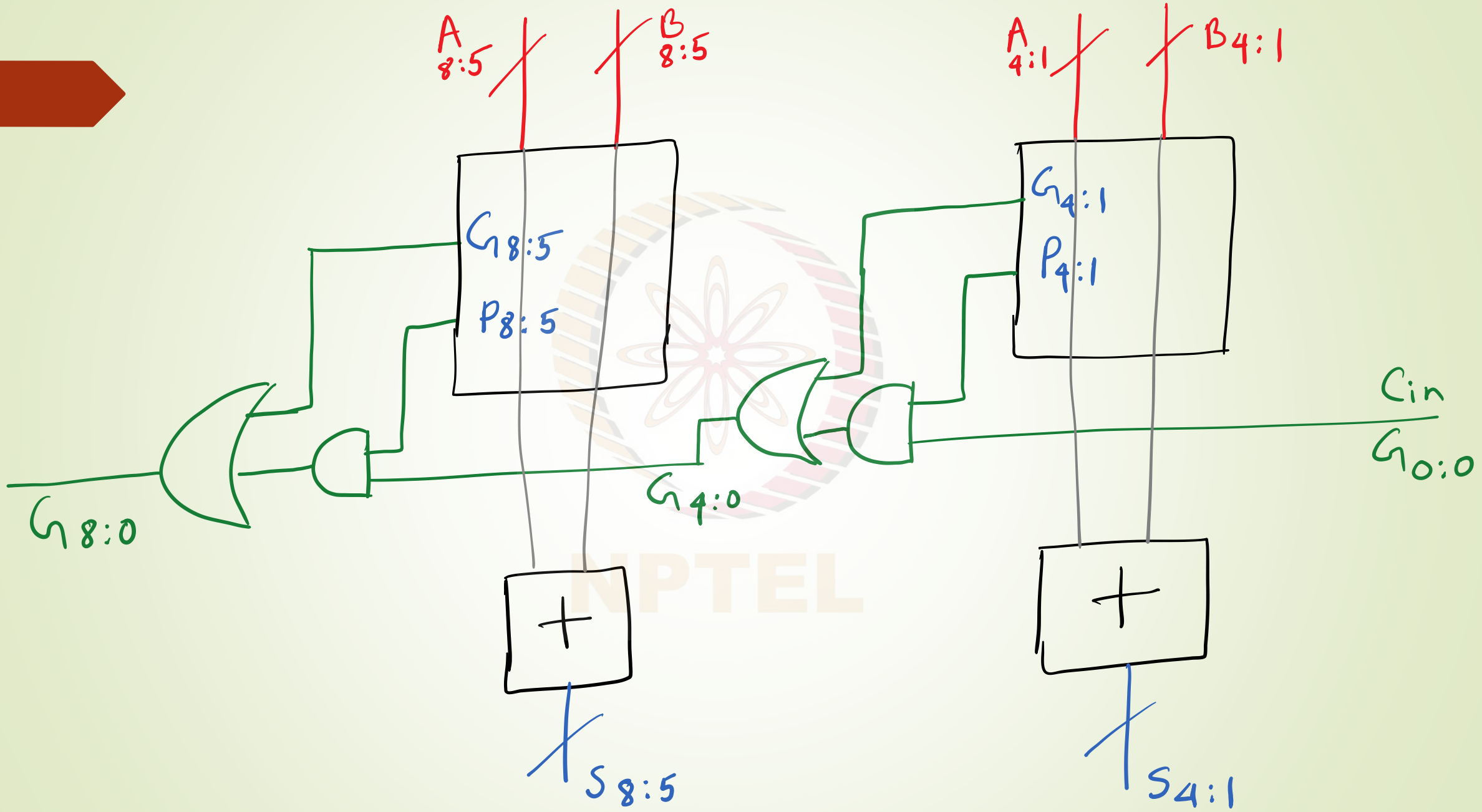
CLA : Same expression as CSA !!

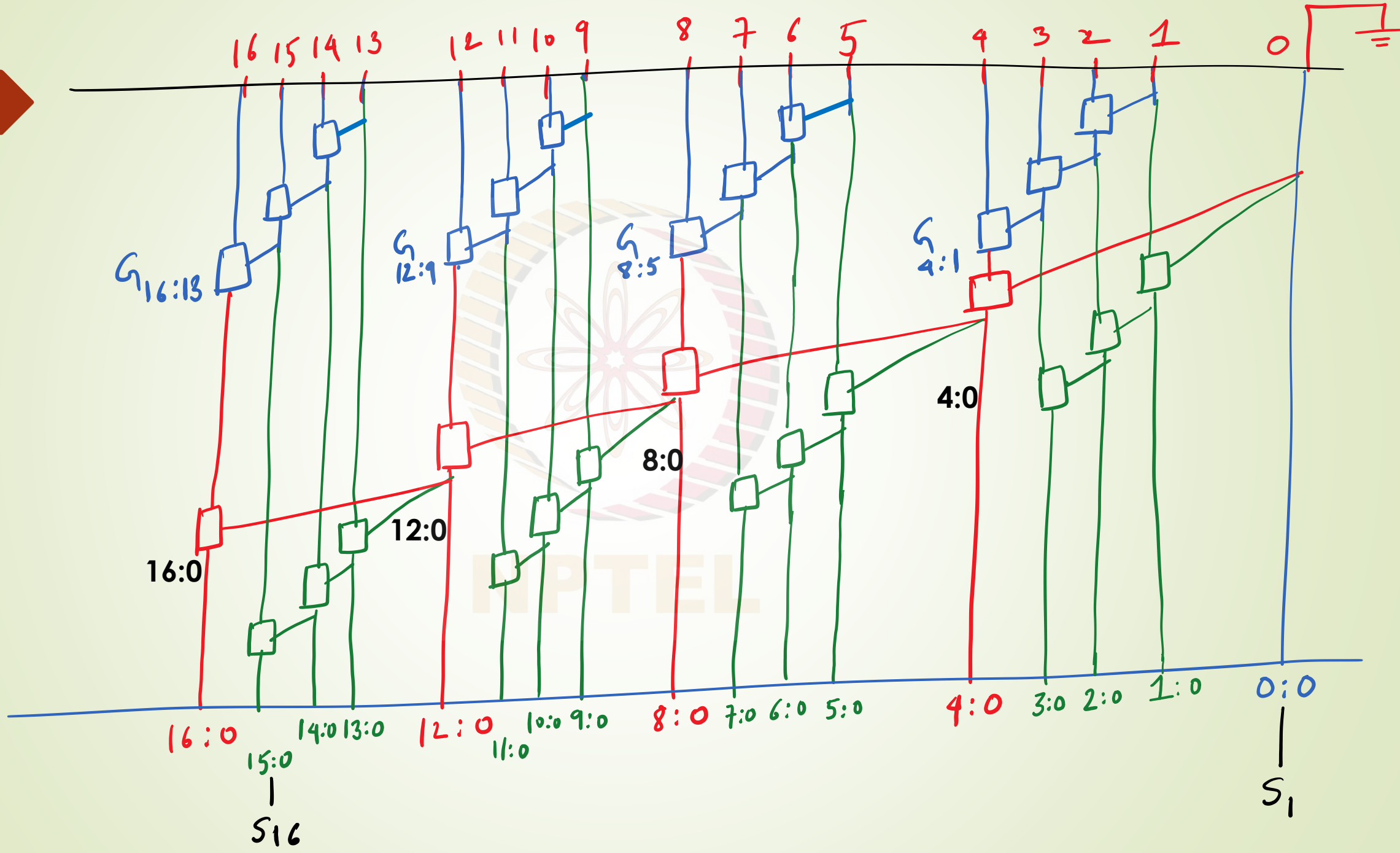
$$G_{4:1} = G_{4:4} + P_{4:4} G_{3:1}$$

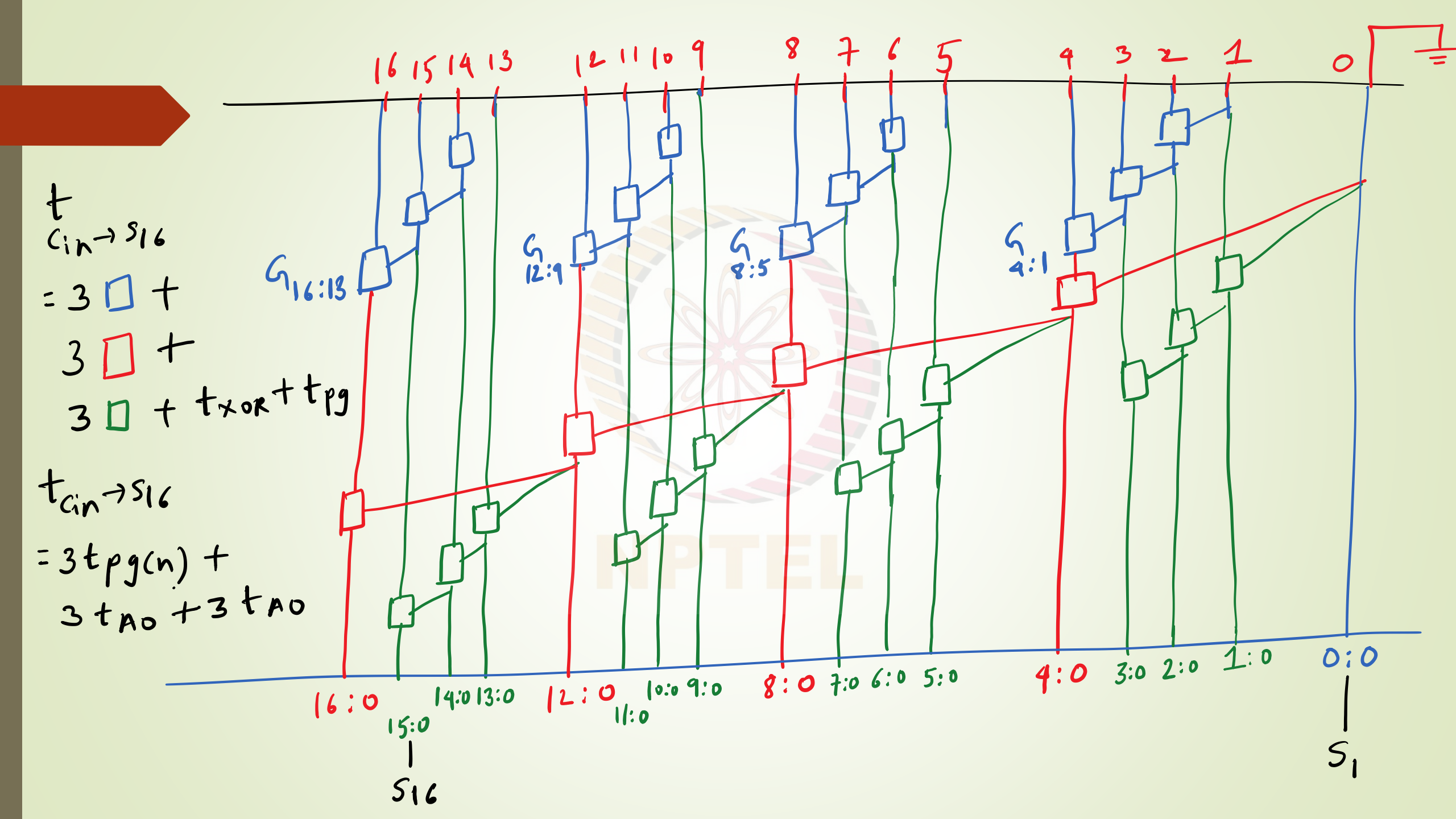
$$G_{3:1} = G_{3:3} + P_{3:3} G_{2:1}$$

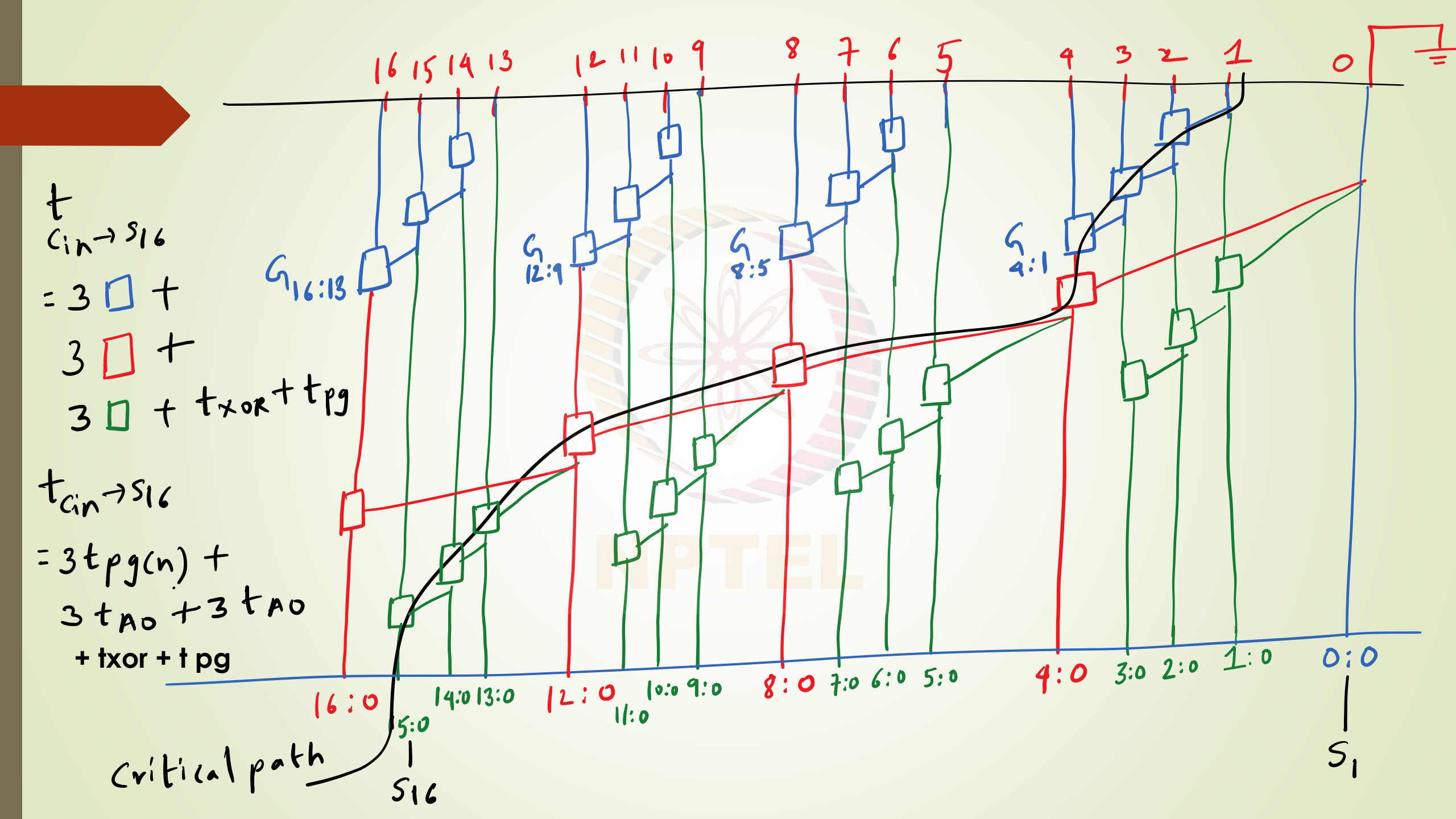
$$G_{2:1} = G_{2:2} + P_{2:2} G_{1:1}$$

$$\boxed{G_{4:0} = G_{4:1} + P_{4:1} G_{0:0}}$$


A 8:5
B 8:5
G 8:5
P 8:5
G 8:0
G 4:0
A 4:1
B 4:1
G 4:1
P 4:1
Cin
G 0:0
S 8:5
S 4:1


16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
G16:13
G12:9
G8:5
G4:1
16:0
12:0
8:0
4:0
16:0 15:0 14:0 13:0 12:0 11:0 10:0 9:0 8:0 7:0 6:0 5:0 4:0 3:0 2:0 1:0 0:0
S16
S1


$t_{C_{in} \rightarrow S_{16}}$
$= 3\square + 3\square + 3\square + t_{XOR} + t_{pg}$
$t_{C_{in} \rightarrow S_{16}}$
$= 3t_{pg(n)} + 3t_{AO} + 3t_{AO}$
16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
$G_{16:13}$
$G_{12:9}$
$G_{8:5}$
$G_{4:1}$
16:0
15:0
14:0
13:0
12:0
11:0
10:0
9:0
8:0
7:0
6:0
5:0
4:0
3:0
2:0
1:0
0:0
S_{16}
S_1


16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
G16:13
G12:9
G8:5
G4:1
16:0
14:0 13:0
12:0
11:0
10:0 9:0
8:0
7:0 6:0 5:0
4:0
3:0 2:0
1:0
0:0
5:0
S16
S1
Critical path
tCin→S16
= 3 □ +
3 □ +
3 □ + txor + tpg
tCin→S16
= 3 tpg(n) +
3 tAO + 3 tAO
+ txor + t pg

$$t_{Cin \rightarrow S_{16}} = t_{pg} + t_{pg(N-1)} + (N-1+k-1)\,t_{AO} + t_{XOR}$$

$(N-1)\,t_{AO}$

k - group of N - bits

Carry Increment Adder

$$G_{8:0} = G_{8:5} + P_{8:5}\,G_{4:0}$$

$$G_{7:0} = G_{7:5} + P_{7:5}\,G_{4:0}$$

$$G_{6:0} = G_{6:5} + P_{6:5}\,G_{4:0}$$

$$G_{5:0} = G_{5:5} + P_{5:5}\,G_{4:0}$$

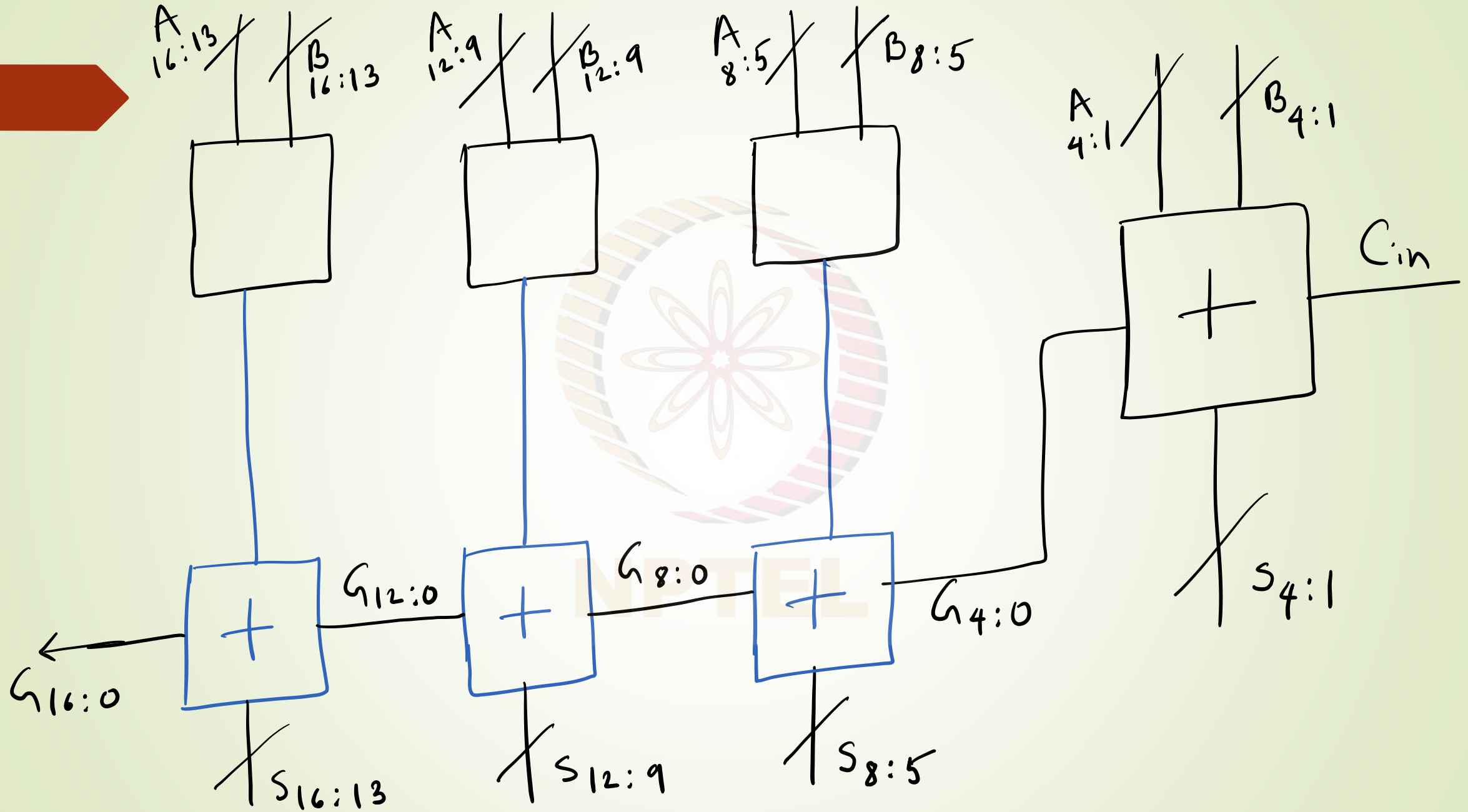
CLA or CSA

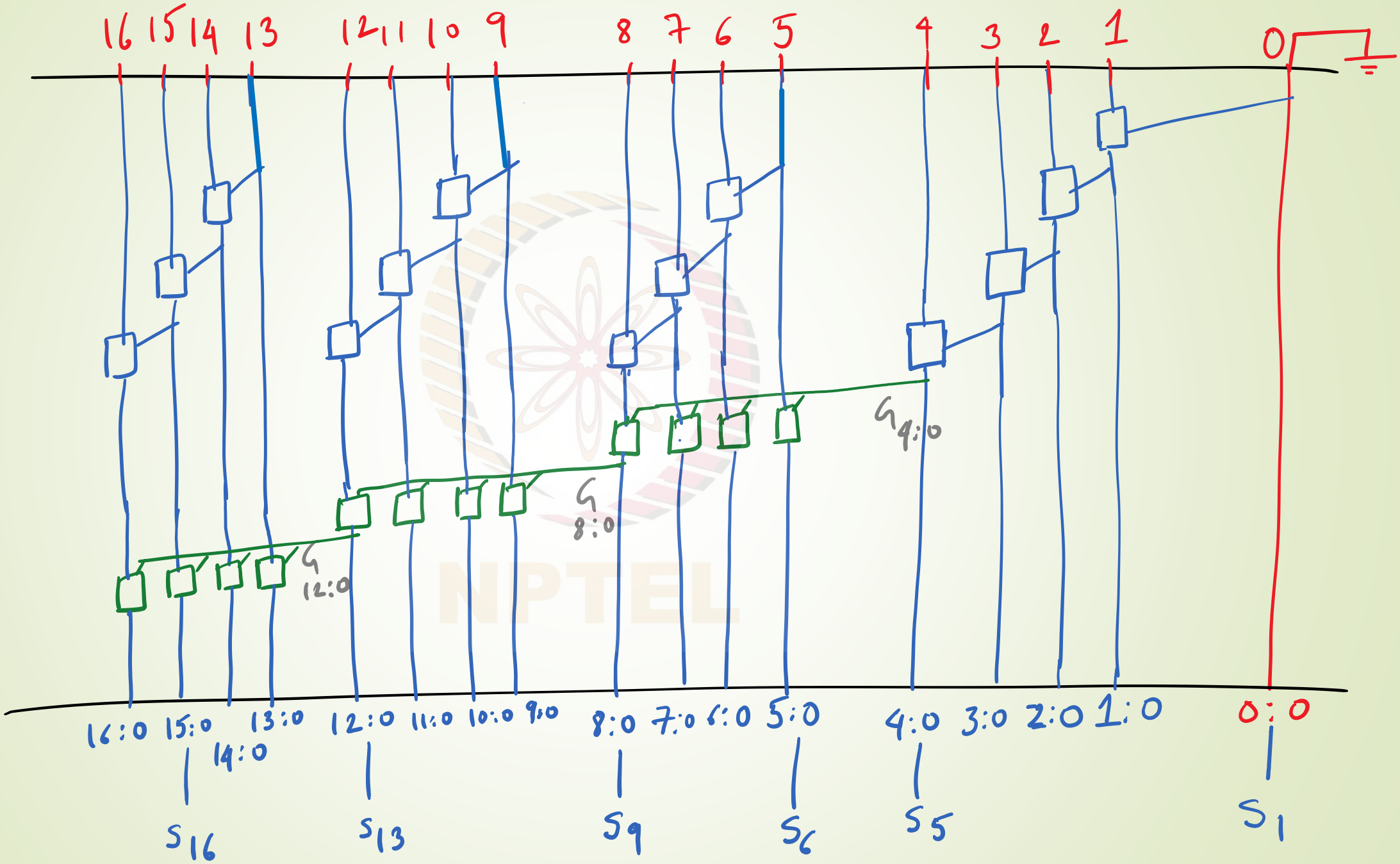
$$G_{8:0} = G_{8:5} + P_{8:5}\,G_{4:0}$$

$$G_{7:0} = G_{7:7} + P_{7:7}\,G_{6:0}$$

$$G_{6:0} = G_{6:6} + P_{6:6}\,G_{5:0}$$

$$G_{5:0} = G_{5:5} + P_{5:5}\,G_{4:0}$$


A 16:13
B 16:13
A 12:9
B 12:9
A 8:5
B 8:5
A 4:1
B 4:1
Cin
G12:0
G8:0
G4:0
G16:0
S16:13
S12:9
S8:5
S4:1


16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
G 12:0
G 8:0
G 4:0
16:0 15:0 14:0 13:0 12:0 11:0 10:0 9:0 8:0 7:0 6:0 5:0 4:0 3:0 2:0 1:0 0:0
S16 S13 S9 S6 S5 S1

Datapath subsystems – Other Adders

Instructor – Dr. Madhav Rao

Carry Increment Adder

$$G_{8:0} = G_{8:5} + P_{8:5}\,G_{4:0}$$

$$G_{7:0} = G_{7:5} + P_{7:5}\,G_{4:0}$$

$$G_{6:0} = G_{6:5} + P_{6:5}\,G_{4:0}$$

$$G_{5:0} = G_{5:5} + P_{5:5}\,G_{4:0}$$

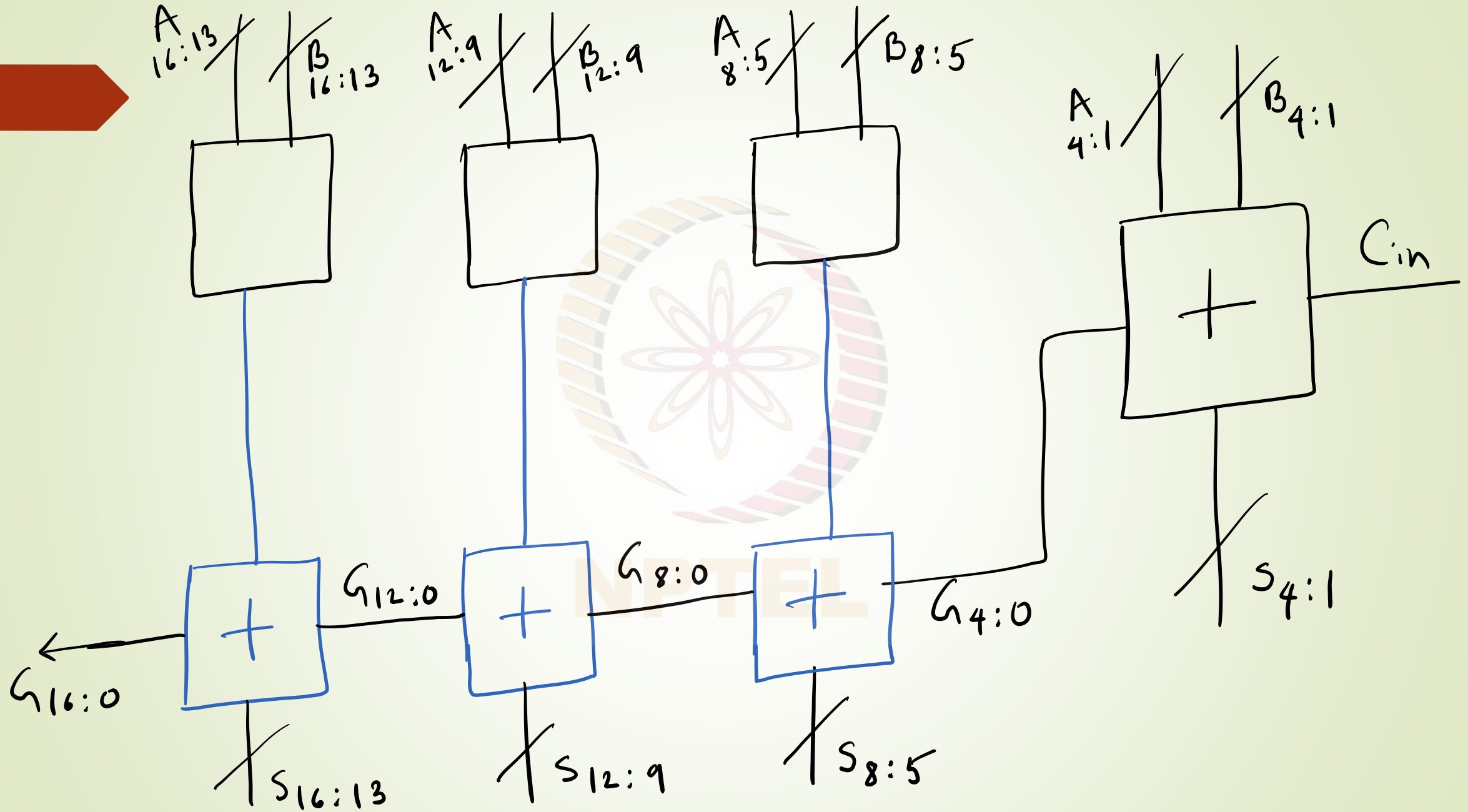
CLA or CSA

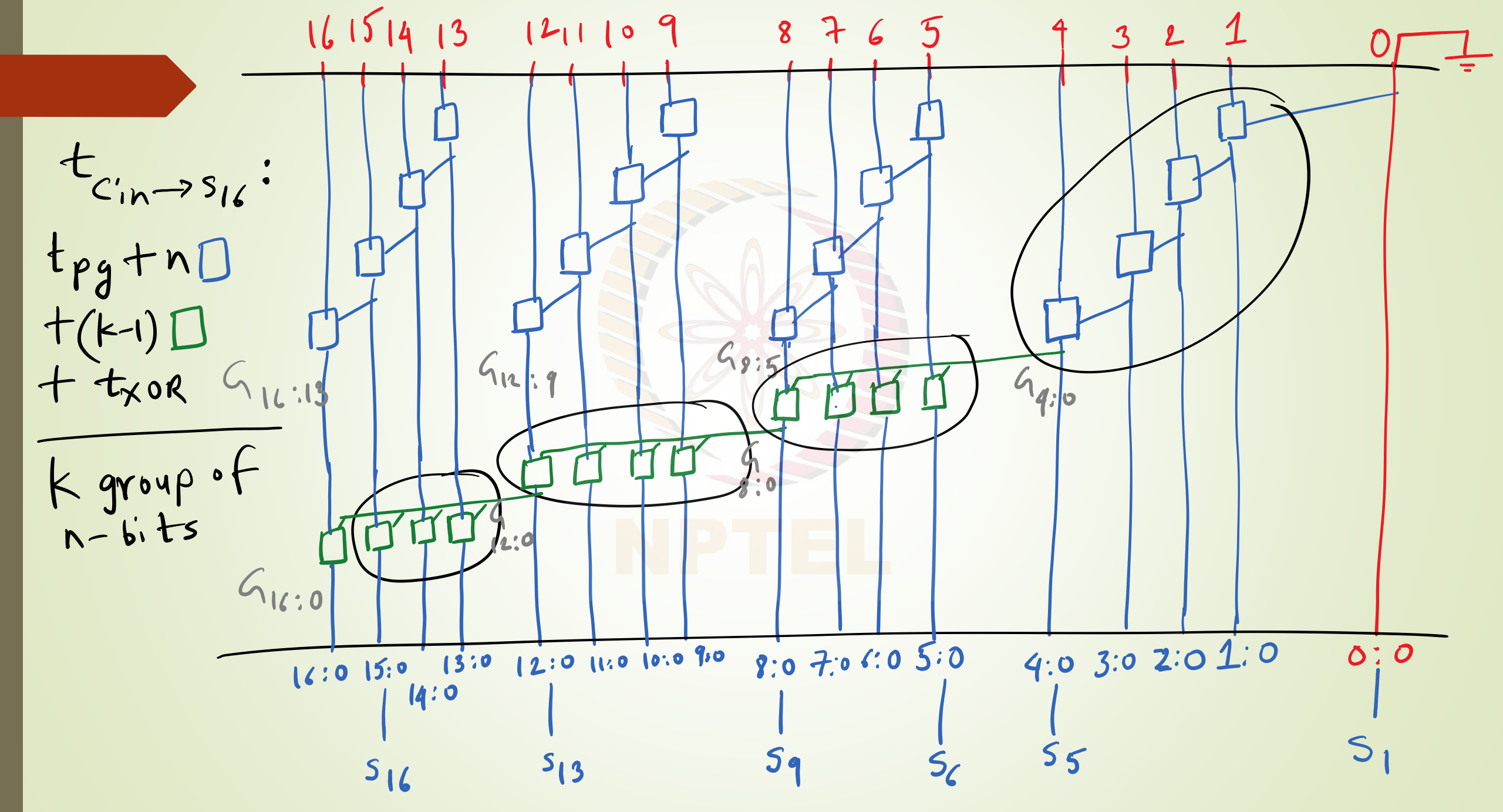
$$G_{8:0} = G_{8:5} + P_{8:5}\,G_{4:0}$$

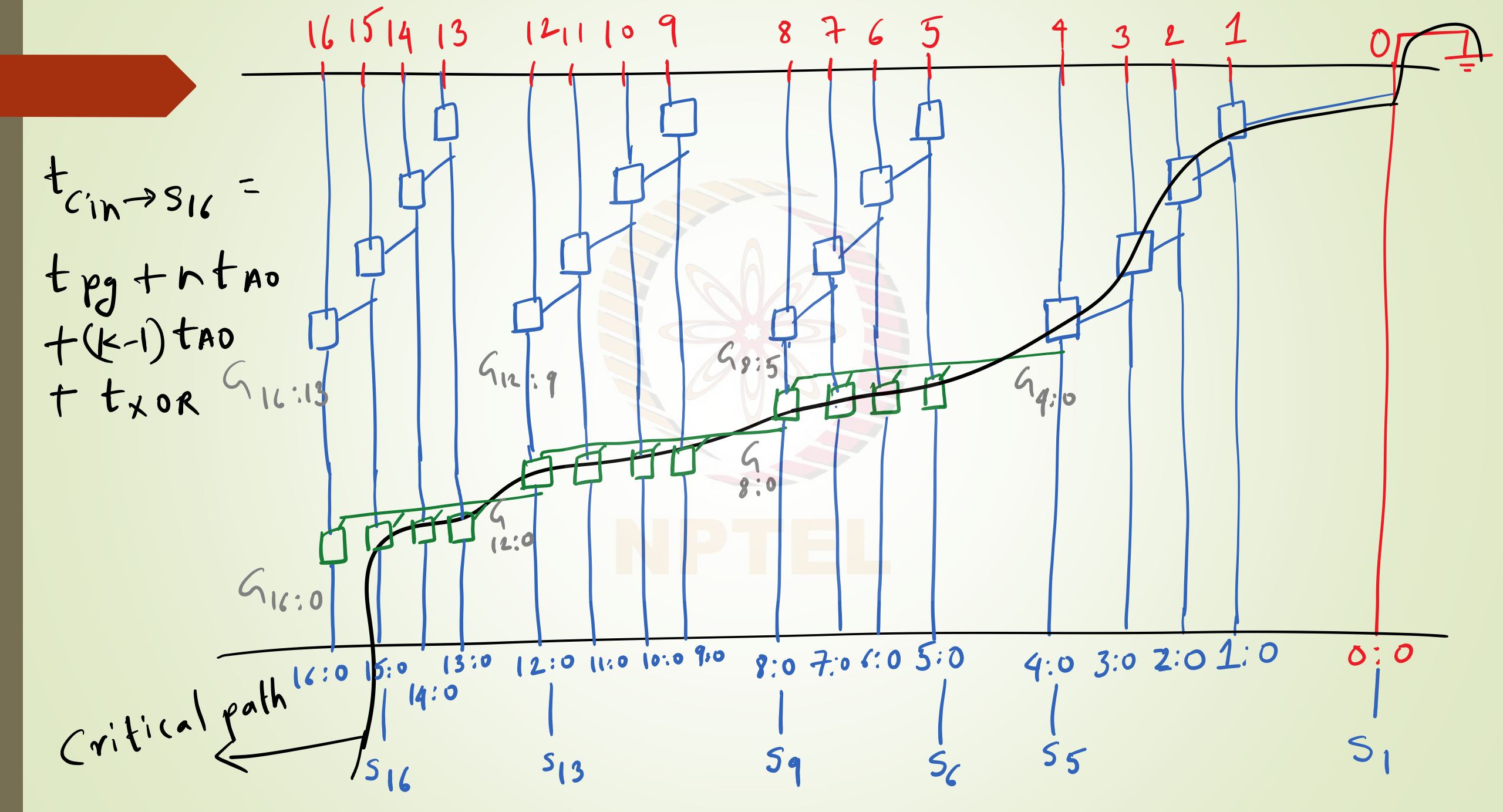
$$G_{7:0} = G_{7:7} + P_{7:7}\,G_{6:0}$$

$$G_{6:0} = G_{6:6} + P_{6:6}\,G_{5:0}$$

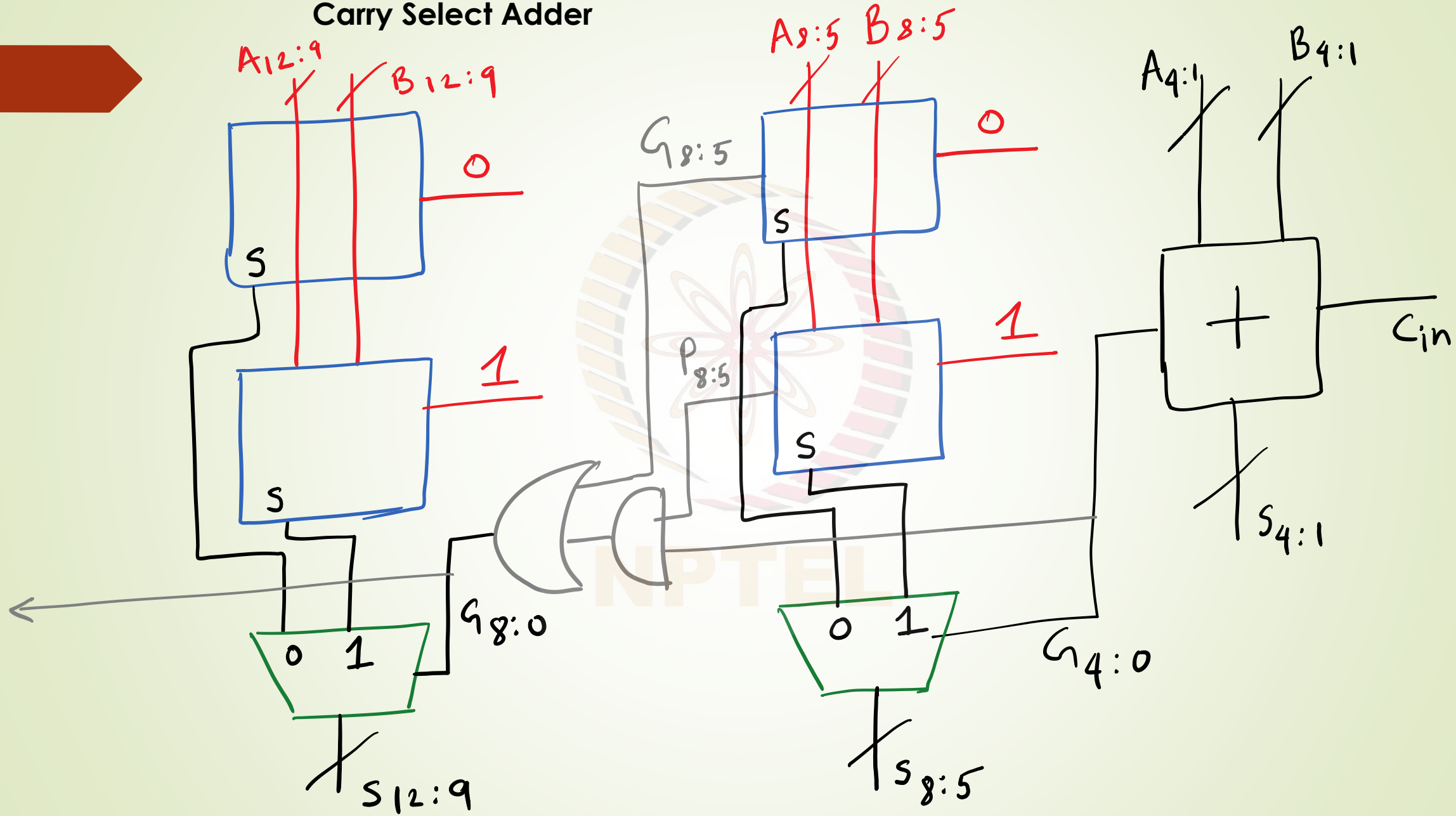
$$G_{5:0} = G_{5:5} + P_{5:5}\,G_{4:0}$$


A 16:13
B 16:13
A 12:9
B 12:9
A 8:5
B 8:5
A 4:1
B 4:1
Cin
G12:0
G8:0
G4:0
G16:0
S16:13
S12:9
S8:5
S4:1


$t_{Cin \rightarrow S_{16}}$:
$t_{pg} + n$
$+ (k-1)$
$+ t_{XOR}$
k group of n-bits
16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
$G_{16:13}$
$G_{12:9}$
$G_{9:5}$
$G_{4:0}$
$G_{8:0}$
$G_{12:0}$
$G_{16:0}$
16:0 15:0 14:0 13:0 12:0 11:0 10:0 9:0 8:0 7:0 6:0 5:0 4:0 3:0 2:0 1:0 0:0
S_{16} S_{13} S_9 S_6 S_5 S_1


$t_{cin \rightarrow s16} =$
$t_{pg} + n t_{AO} + (k-1) t_{AO} + t_{XOR}$
16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
$G_{16:13}$
$G_{12:9}$
$G_{9:5}$
$G_{4:0}$
$G_{8:0}$
$G_{12:0}$
$G_{16:0}$
16:0 15:0 14:0 13:0 12:0 11:0 10:0 9:0 8:0 7:0 6:0 5:0 4:0 3:0 2:0 1:0 0:0
Critical path
s_{16} s_{13} s_9 s_6 s_5 s_1

Carry Select Adder

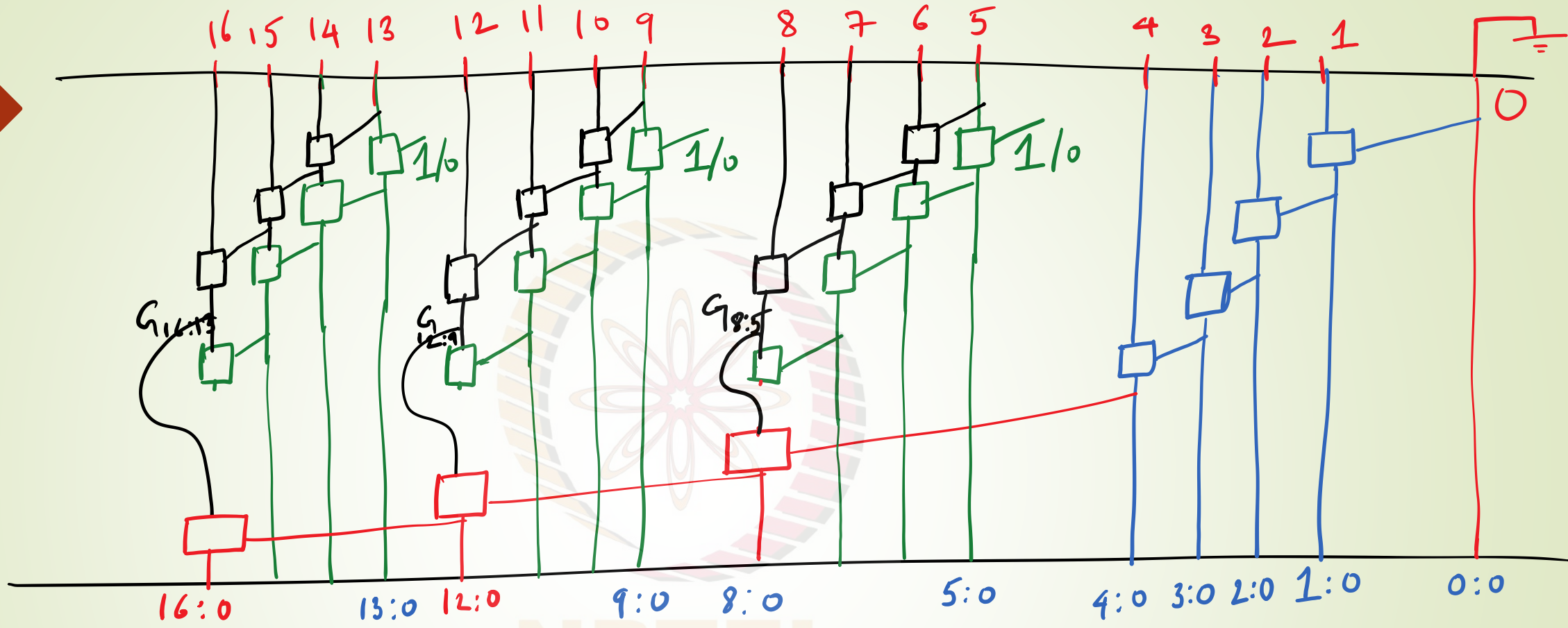


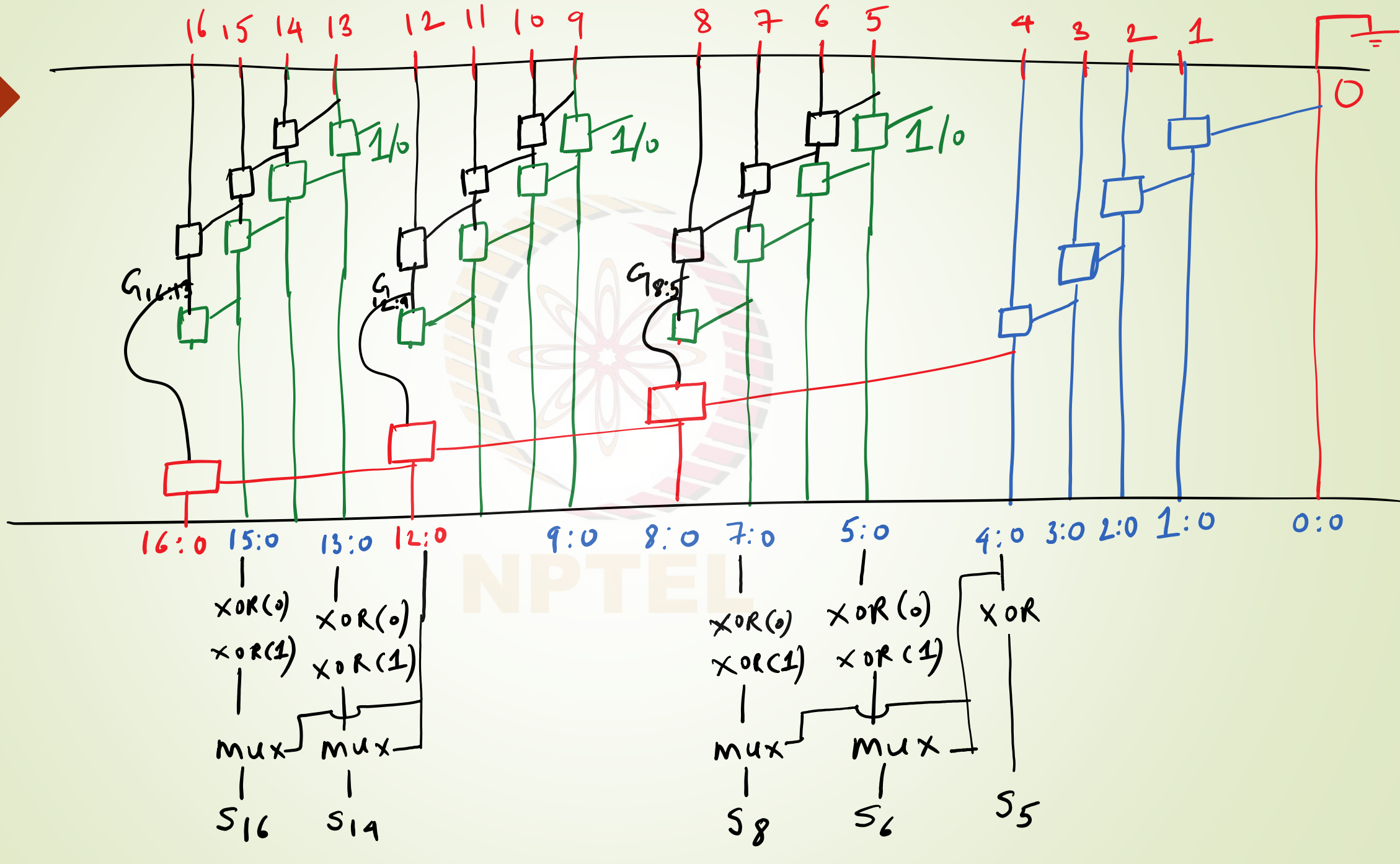
$$G_{i:j} = G_{i:k} + P_{i:k} G_{k-1:j}$$

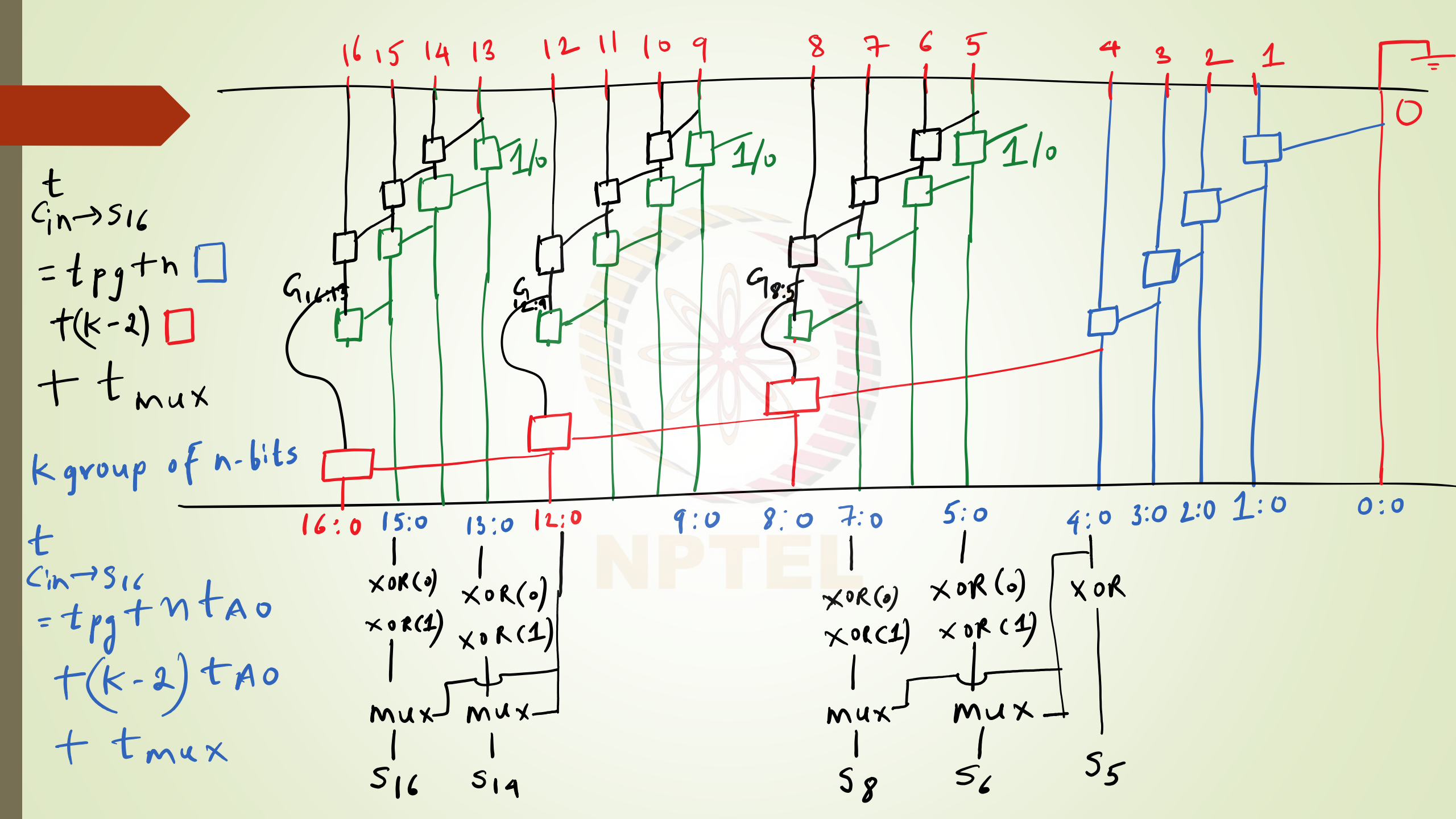
previously generated blocks

Generated by block which takes carry '0'

Generated by block which takes carry '1'


16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
0
1/0
1/0
1/0
G16:13
G12:9
G8:5
16:0
13:0
12:0
9:0
8:0
5:0
4:0
3:0
2:0
1:0
0:0

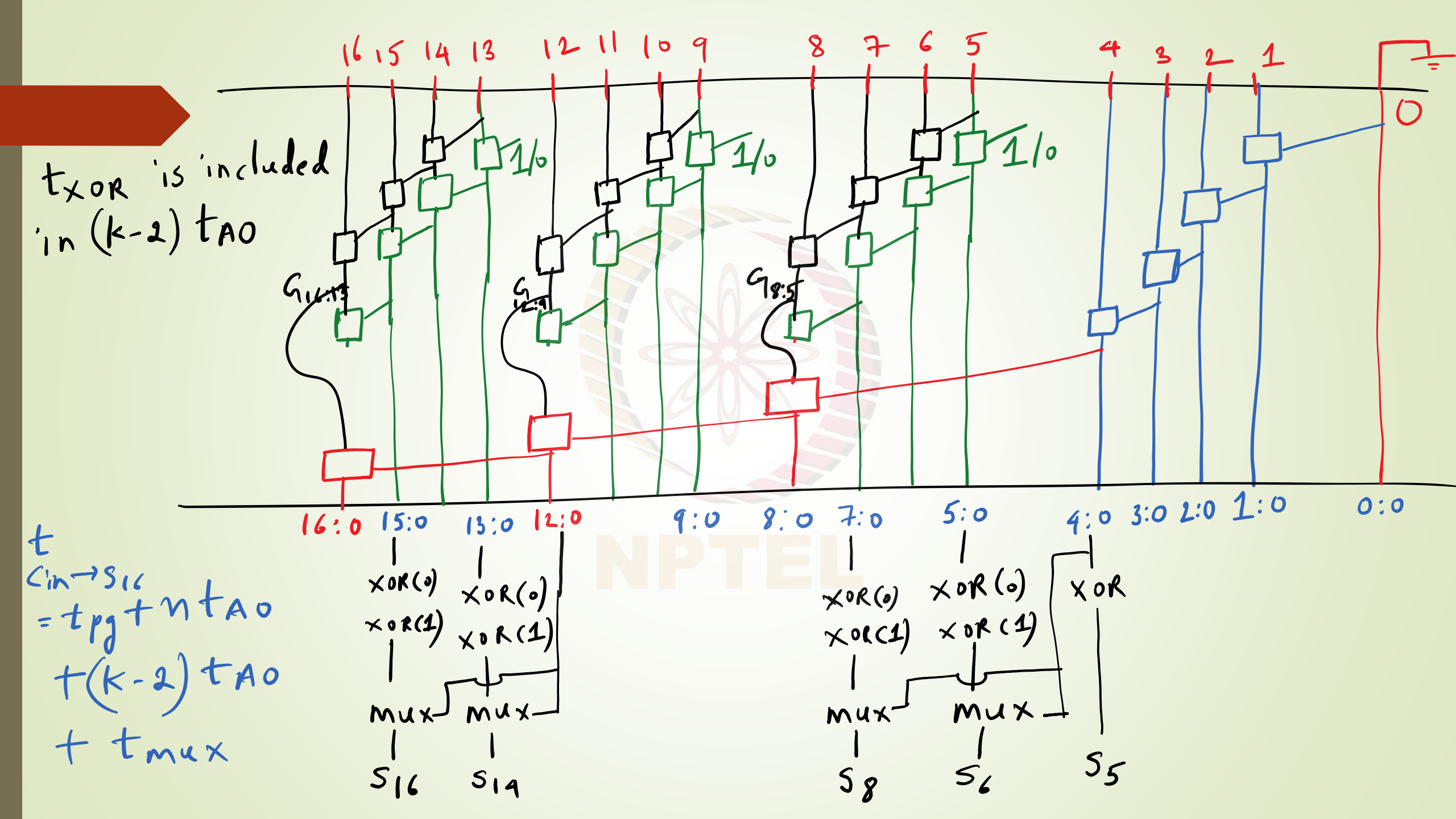

16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
1/0
1/0
1/0
0
G16:13
G12:9
G8:5
16:0 15:0 13:0 12:0 9:0 8:0 7:0 5:0 4:0 3:0 2:0 1:0 0:0
XOR(0)
XOR(1)
XOR(0)
XOR(1)
XOR(0)
XOR(1)
XOR(0)
XOR(1)
XOR
mux
mux
mux
mux
S16
S14
S8
S6
S5



$t_{C_{in} \to S_{16}} = t_{pg} + n \square + t(k-2) \square + t_{mux}$

k group of n-bits

$t_{C_{in} \to S_{16}} = t_{pg} + n t_{AO} + t(k-2) t_{AO} + t_{mux}$


t_{XOR} is included in $(k-2)\,t_{AO}$
16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
1/0
1/0
1/0
0
$G_{16:13}$
$G_{12:9}$
$G_{8:5}$
16:0 15:0 13:0 12:0 9:0 8:0 7:0 5:0 4:0 3:0 2:0 1:0 0:0
XOR(0)
XOR(1)
XOR(0)
XOR(1)
XOR(0)
XOR(1)
XOR(0)
XOR(1)
XOR
mux
mux
mux
mux
S_{16}
S_{14}
S_8
S_6
S_5
$t_{C_{in} \to S_{16}} = t_{pg} + n\,t_{AO} + t(k-2)\,t_{AO} + t_{mux}$