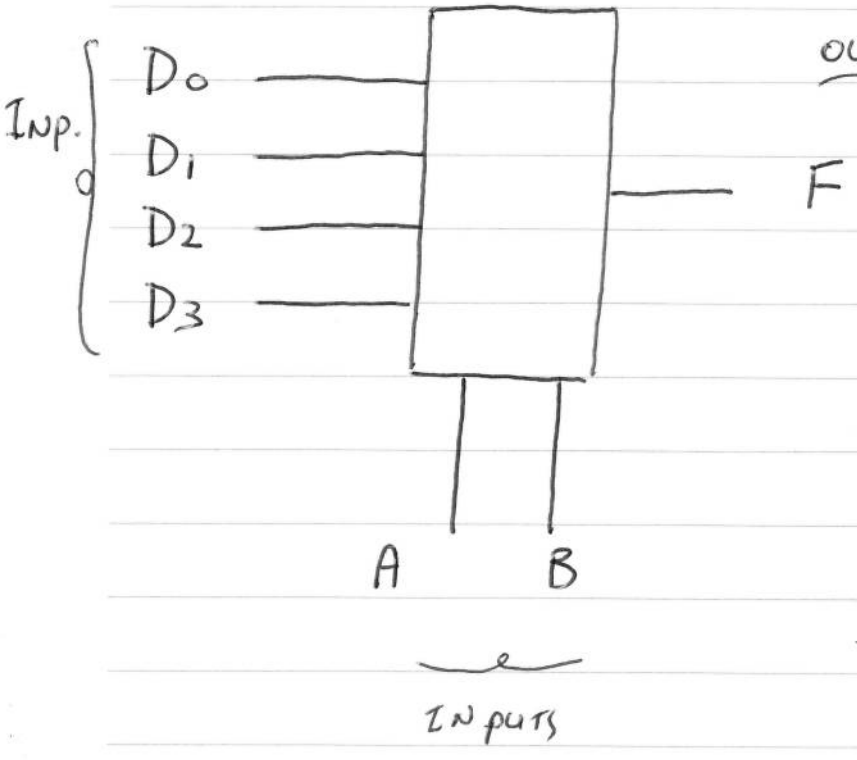
LAST, WE LOOKED AT 2×1 MULTIPLEXER.
CAN ALSO HAVE 4×1 "
8×1 "

4 × 1 MULTIPLEXER



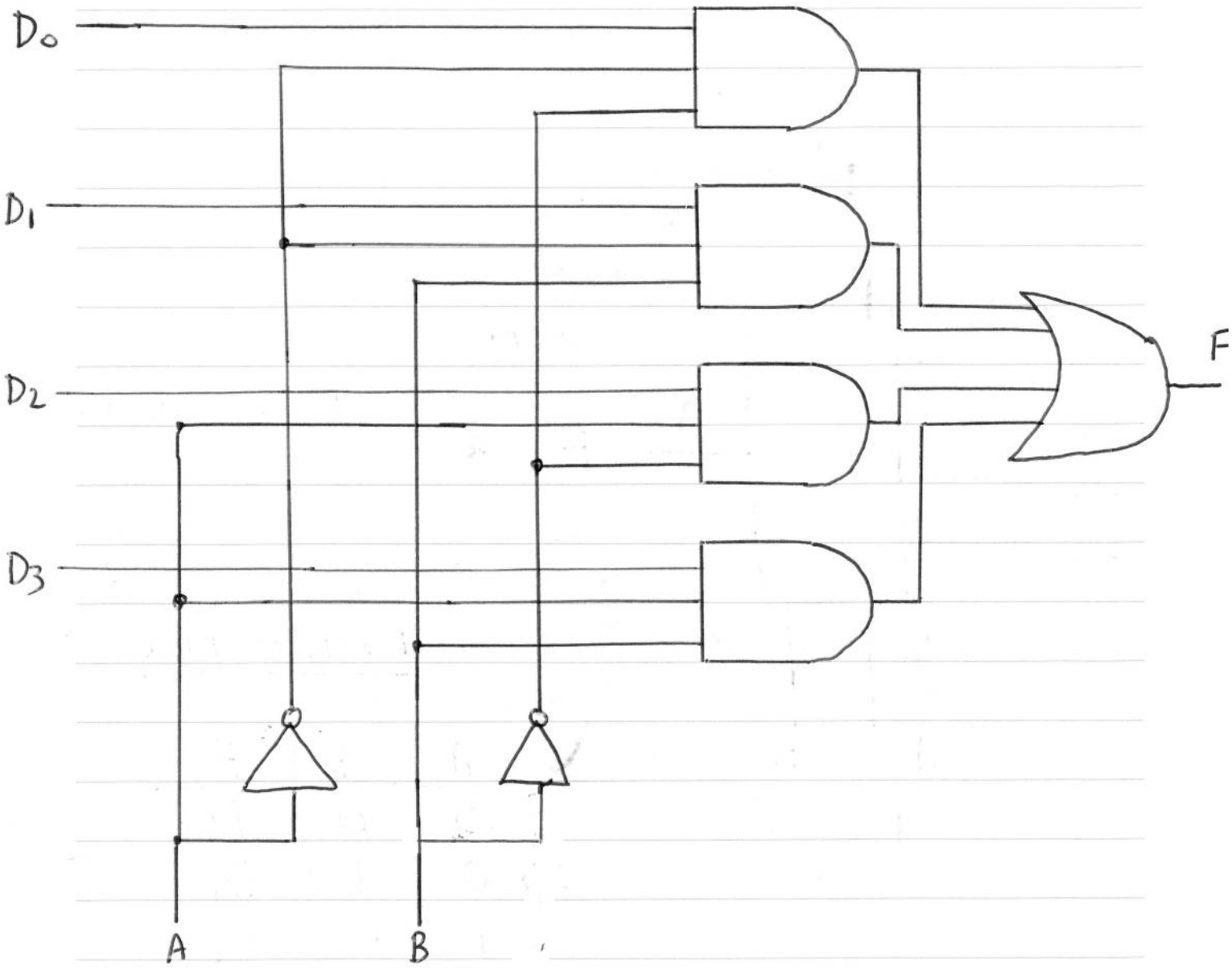
BLOCK DIAGRAM

FUNCTION TABLE

A	B	F
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

$$F = \bar{A}\bar{B}D_0 + \bar{A}BD_1 + A\bar{B}D_2 + ABD_3$$

(2)

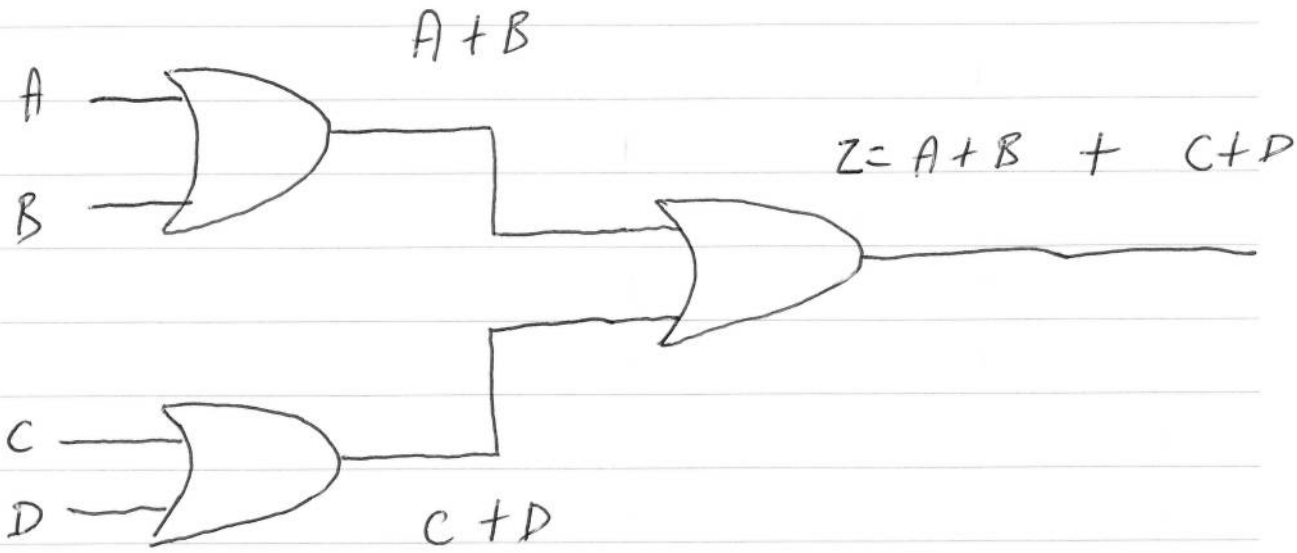


TRUTH TABLE 4 INPUT OR

V	W	X	Y	Z
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

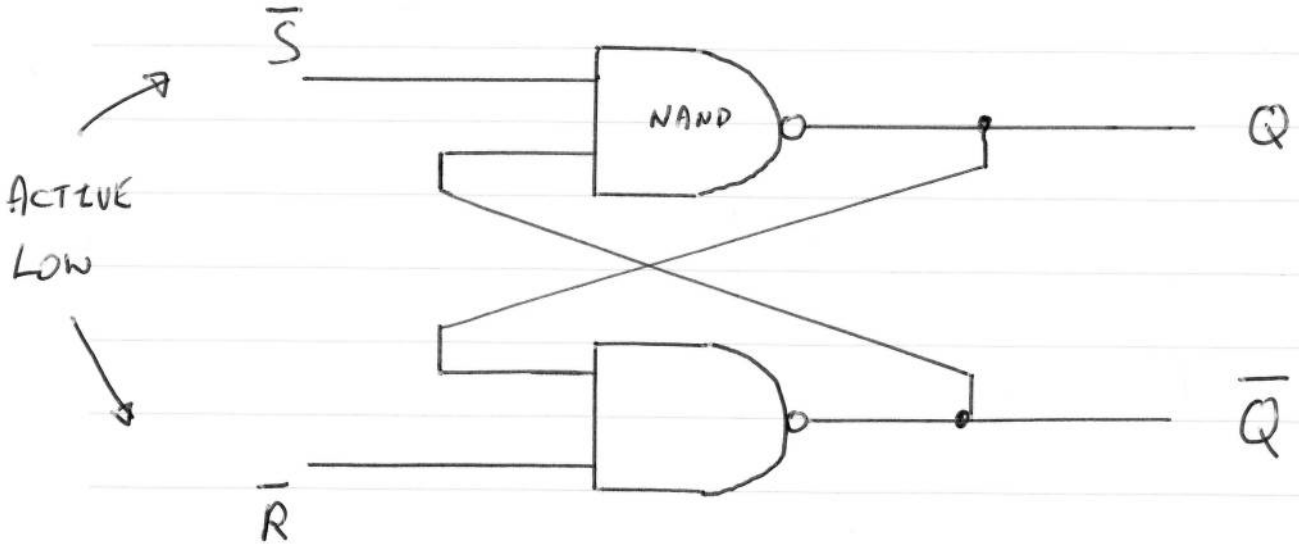
④

HOW WOULD YOU IMPLEMENT A 4 INPUT OR USING 2 INPUT OR GATES?



S-R LATCH

- USES FEEDBACK



NAND

A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

TRUTH TABLE S-R LATCH

$\overline{S}$	$\overline{R}$	Q	$\overline{Q}$
0	0	INVALID	
0	1	1	0
1	0	0	1
1	1	(NO CHANGE) MEMORY STATE	

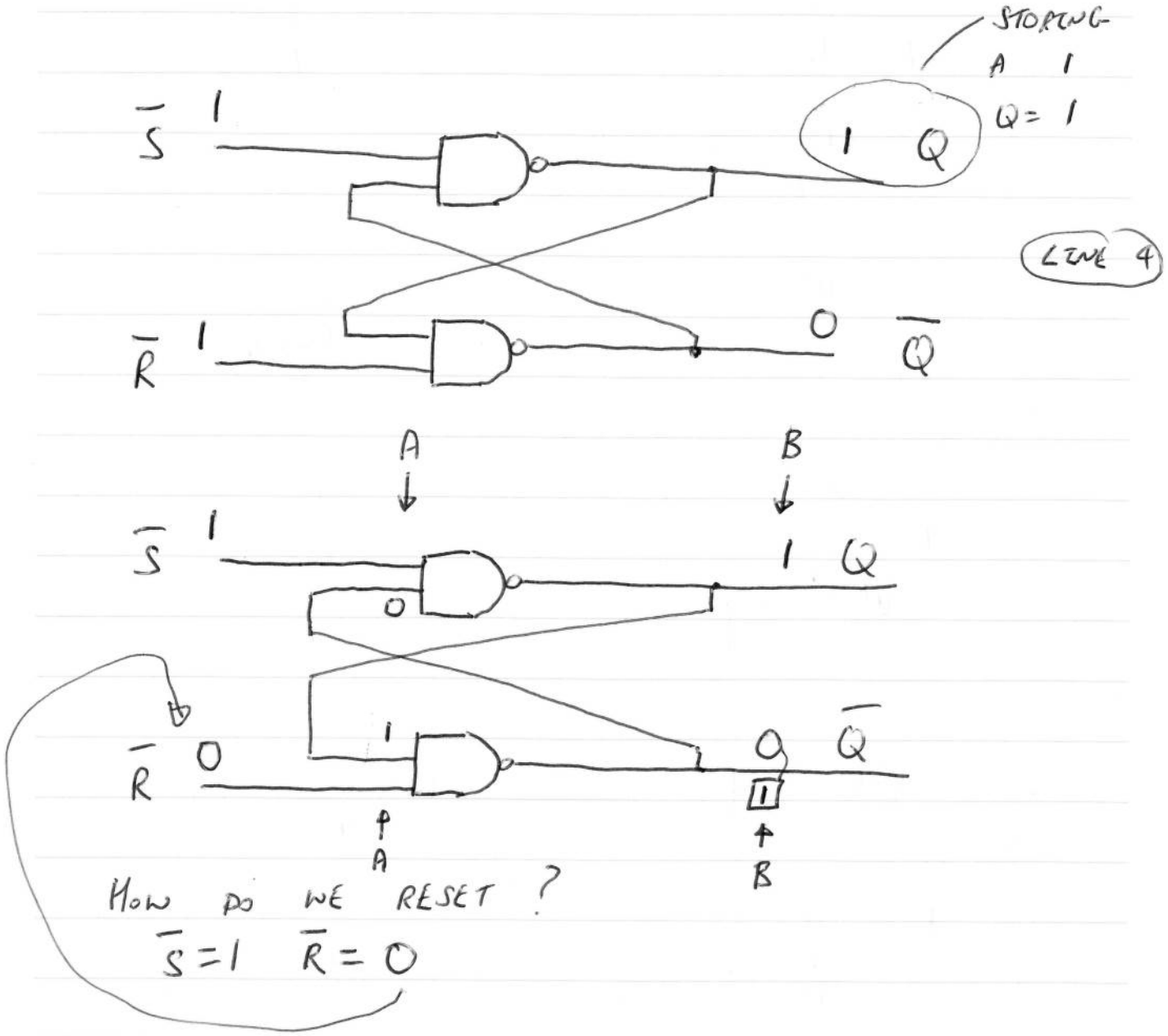
PURPOSE : STORE A SINGLE BIT : 0 OR 1

TO SET	$Q = 1$
$\bar{S} = 0$	$\bar{R} = 1$

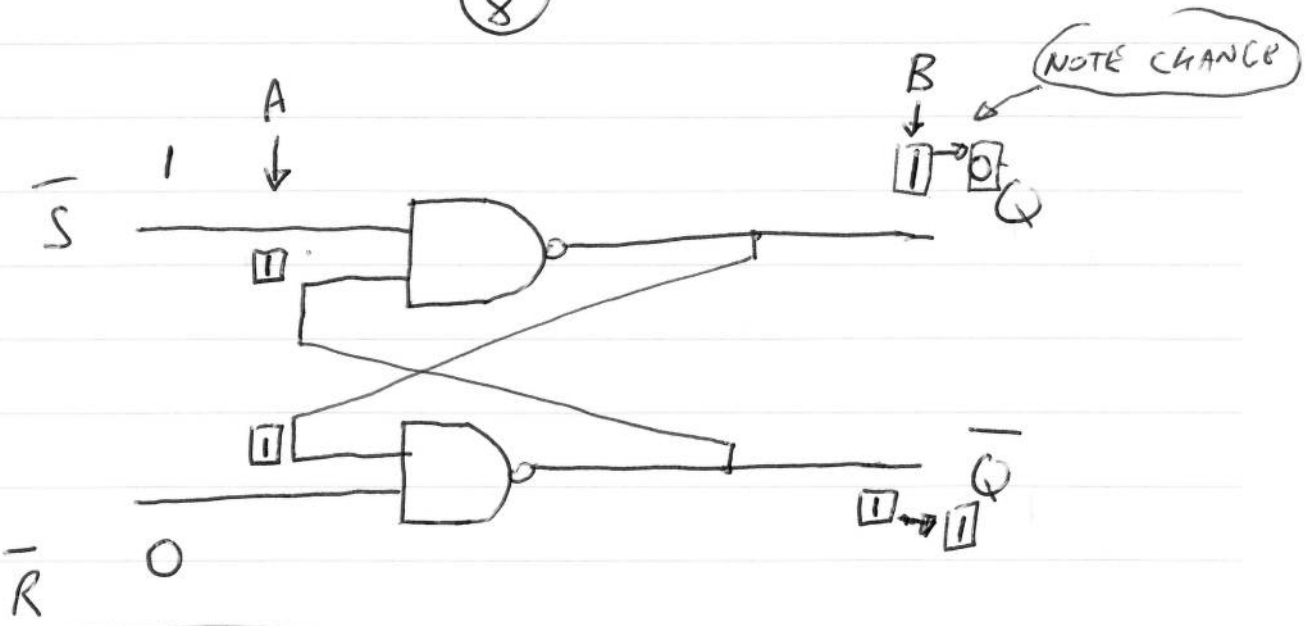
SET

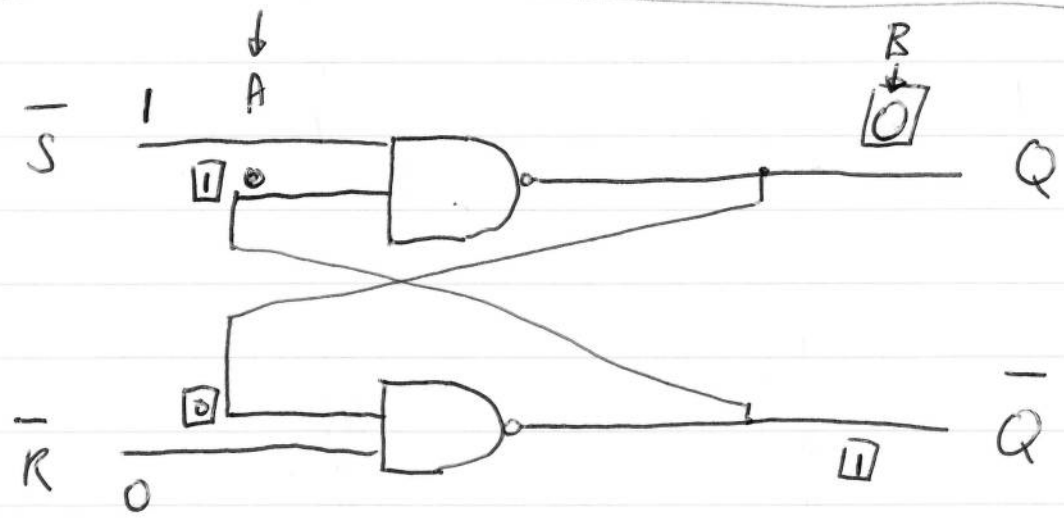
TO RESET	$Q = 0$
$\bar{S} = 1$	$\bar{R} = 0$

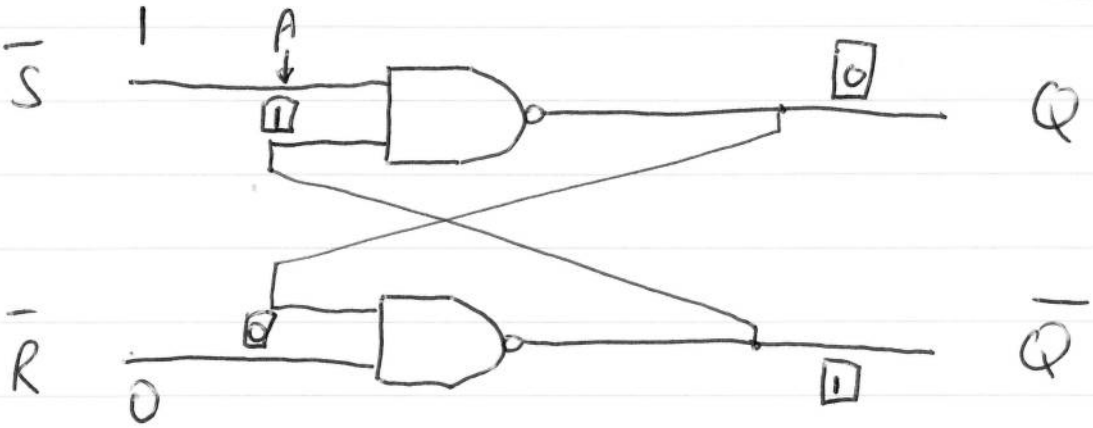
RESET


STORING
A 1
Q = 1
$\bar{S}$ 1
1 Q
LINE 4
$\bar{R}$ 1
0 $\bar{Q}$
A
B
$\bar{S}$ 1
1 Q
0
$\bar{R}$ 0
1
Q $\bar{Q}$
A
B
HOW DO WE RESET ?
$\bar{S}$ = 1 $\bar{R}$ = 0

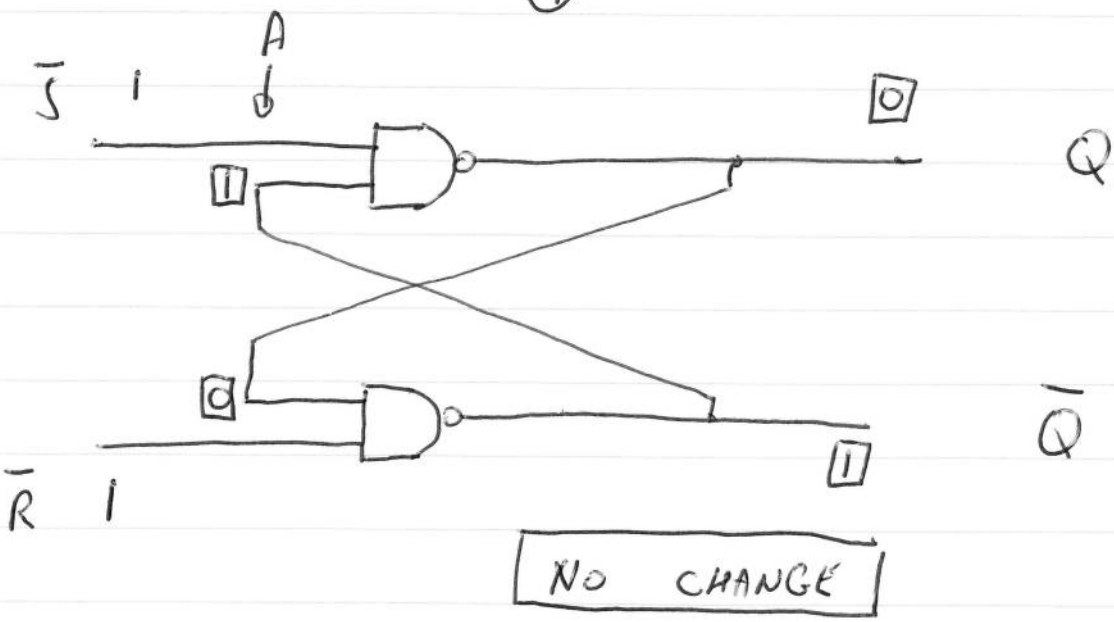
(8)


NOTE CHANGE
B
A
1
$\overline{S}$
1
1
0
Q
1
$\overline{Q}$
1
1
$\overline{R}$
0


A
B
1
$\overline{S}$
1
0
Q
0
$\overline{Q}$
$\overline{R}$
0
1


1
A
$\overline{S}$
1
0
Q
0
$\overline{Q}$
$\overline{R}$
0
1

(9)



NO CHANGE