```
use.ieee.std_logic_1164.all;
USE.ieee.std_logic_UNSIGNED.all;
ENTITY ALU IS
PORT( S: IN STD_LOGIC_VECTOR (2 DOWN TO 0);
      A,B: IN STD_LOGIC_VECTOR(3 DOWNTO 0);
      F: OUT STD_LOGIC_VECTOR(3 DOWNTO 0));
END ALU;
ARCHITECTURE Behavior OF ALU IS
BEGIN
 PROCESS (S, A, B)
 BEGIN
 CASE IS
```

```
WHEN "000" => F <= "0000" ; --CLEAR
WHEN "001" => F <= B-A ;
        :
WHEN "110" => F <= A AND B ;
WHEN OTHERS => F <= "1111" ;
END CASE;
END PROCESS;
END BEHAVIOR;
```

For logic operations

bit-by-bit

$a_k \cdot b_k = f_k$

$a_0 \cdot b_0 = f_0$

$a_1 \cdot b_1 = f_1$

$a_2 \cdot b_2 = f_2$

$a_3 \cdot b_3 = f_3$

$\downarrow$

$f_3 f_2 f_1 f_0$

$S \Leftarrow A+B+C+D$

in arithmetic operation (addition)

precedence:

$((A+B)+C)+D=S$

1 operation

1 operation

1 operation

3 operations

$S=(a+b)+(c+d)$

operation 1

operation 2

operation 3 ⇒ 3 operations

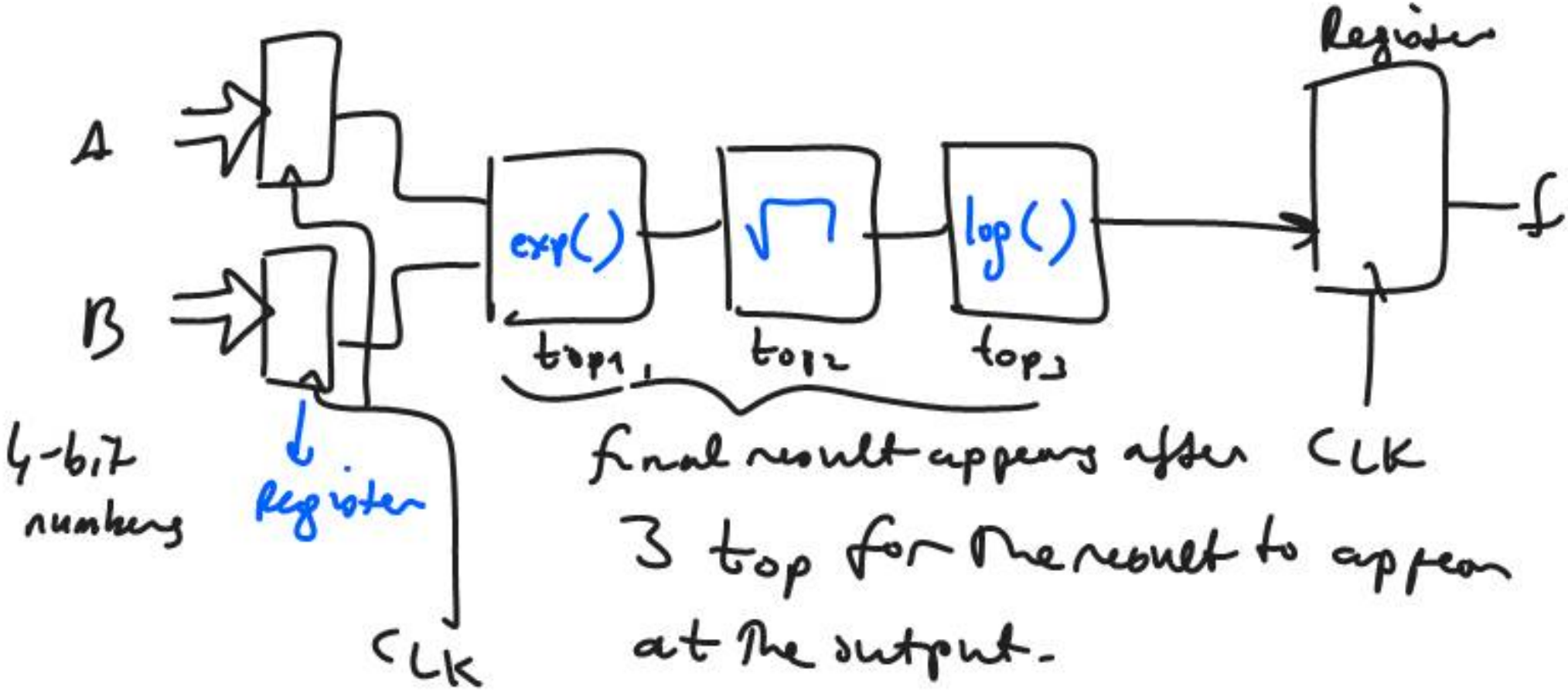
Cascaded 3 adders

3 tp

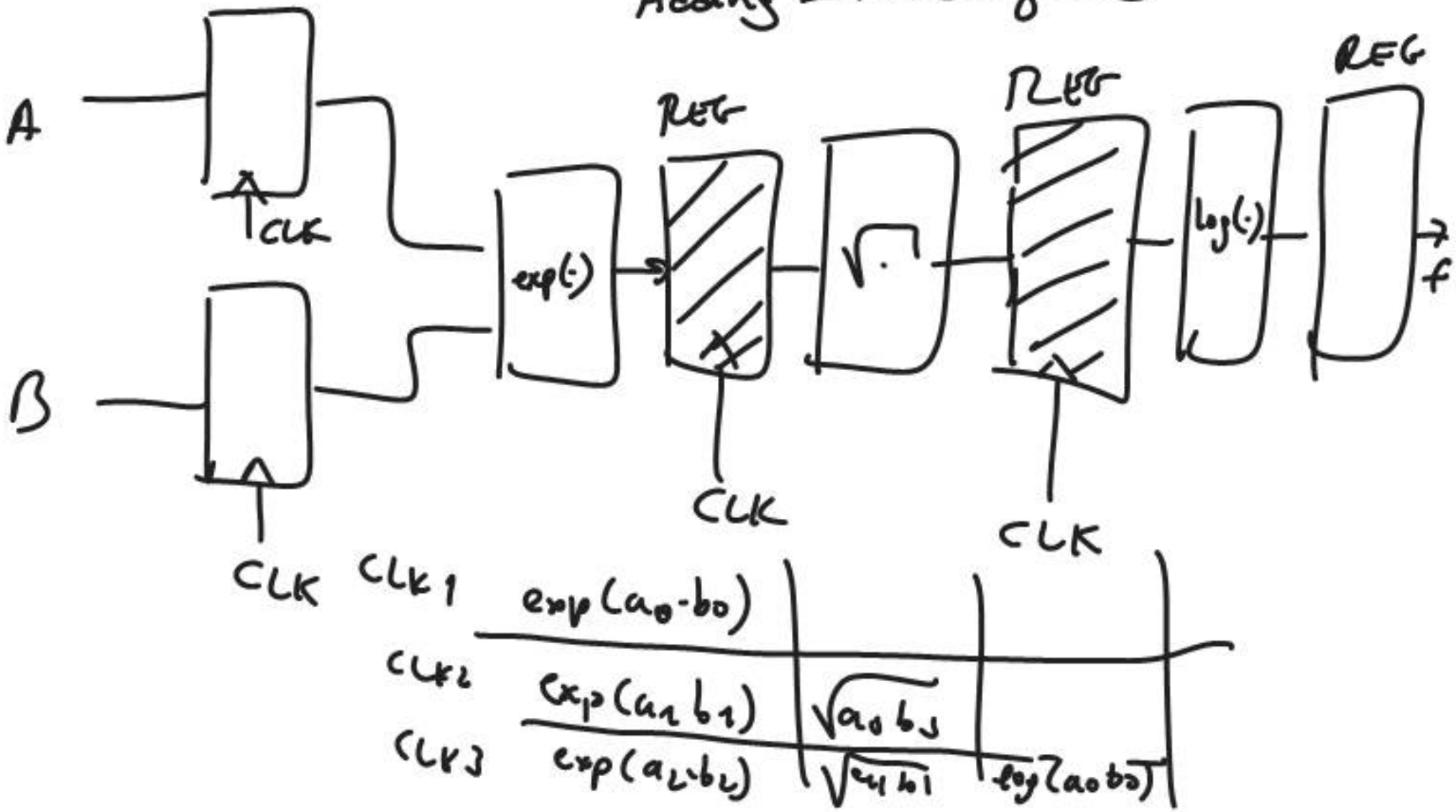
2 tp

propagation times!

Or we can use PIPELINE approach:

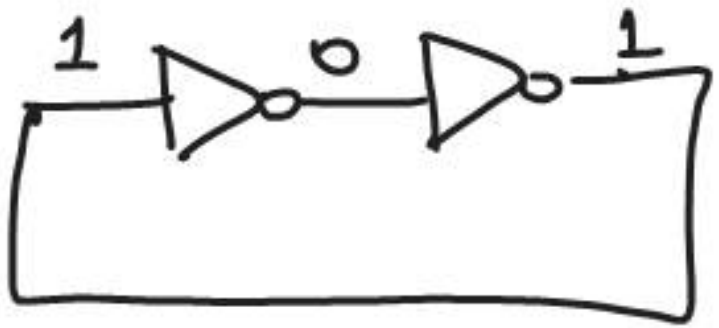


If we use pipelining → Adding 2 more registers!



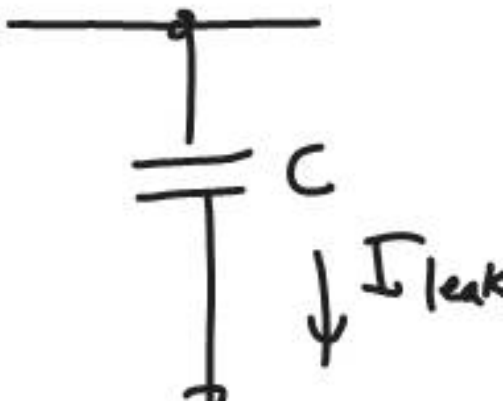
SEQUENTIAL CIRCUITS

A simple memory element



A BISTABLE circuit
it keeps 1 bit of information

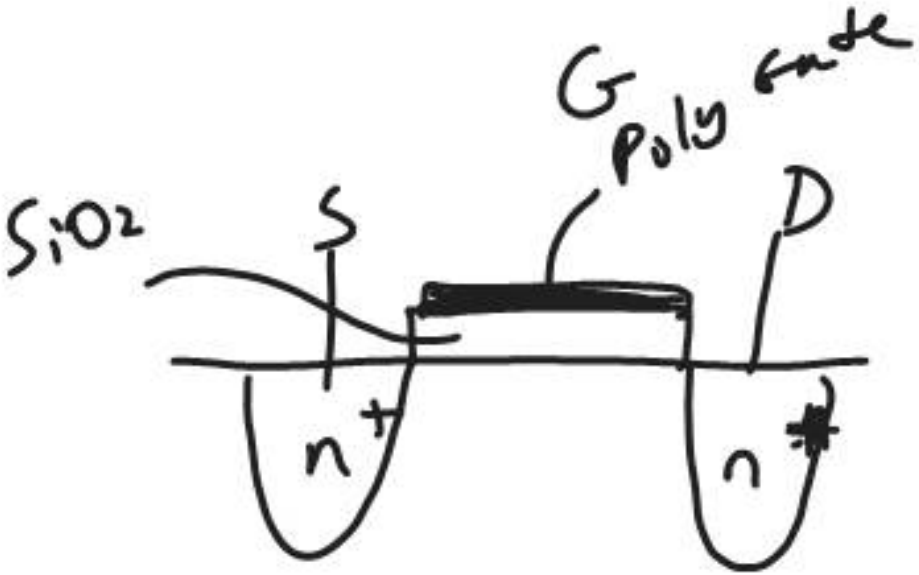
STATIC memory

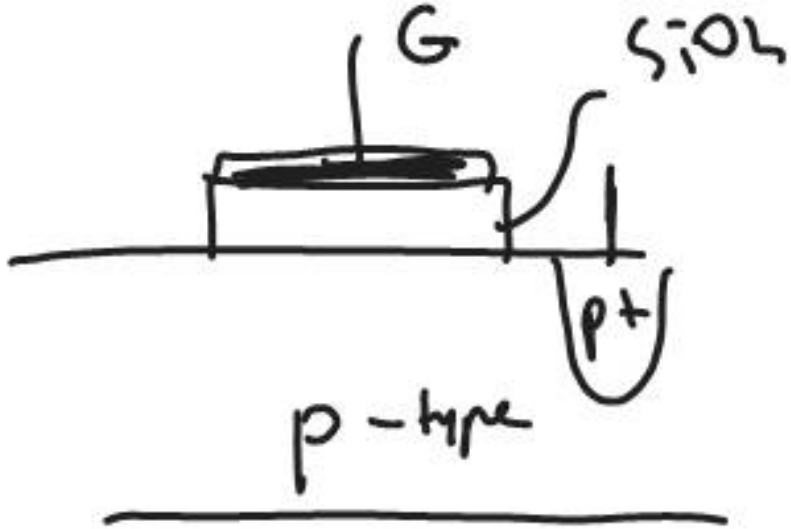


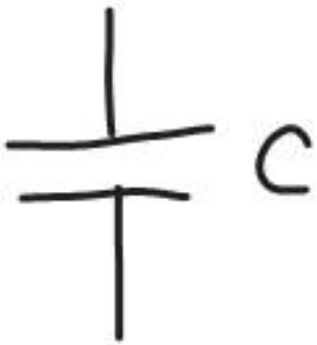
It should be refreshed continously

DYNAMIC MEMORY

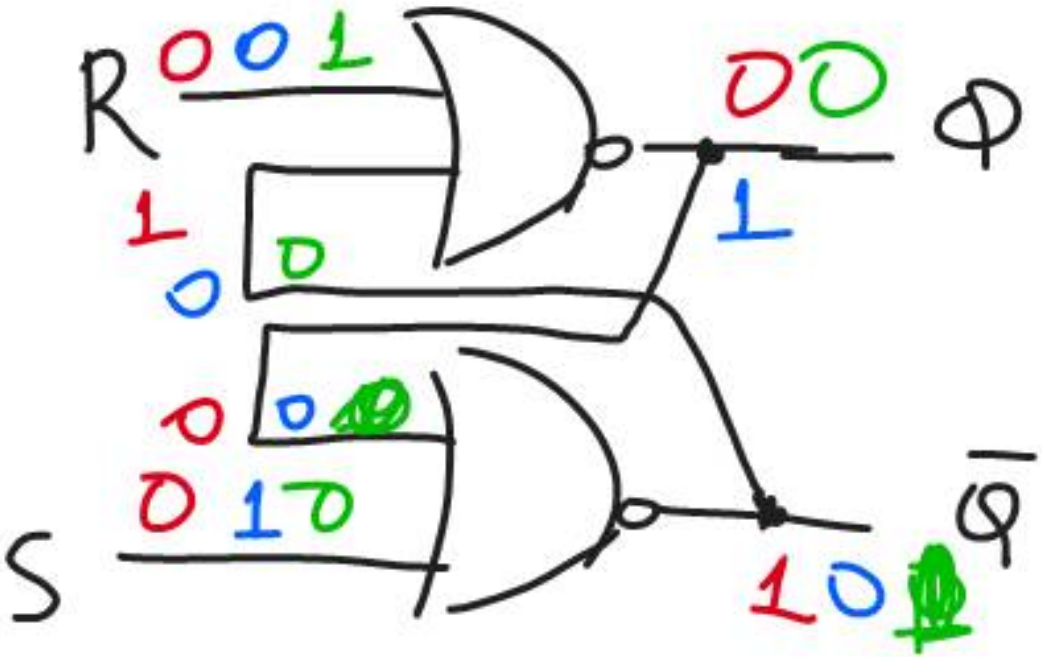
VOLATILE







n-channel
CMOS transistor



x	y	f
0	0	1
0	1	0
1	0	0
1	1	0

NOR GATE

SR LATCH

Characteristic Table

S	R	Q
0	0	No change
0	1	0
1	0	1
(1)	(1)	May oscillate