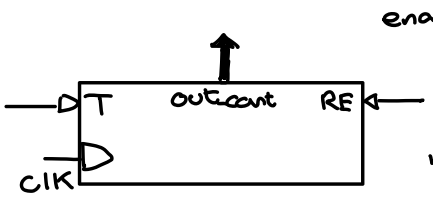
Digital Counter



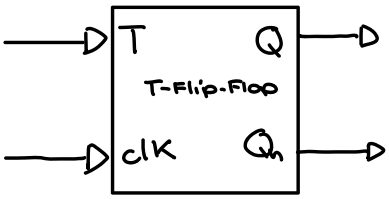
enable

Signal Name	Direction	Resolution
T	In	1 bit
reset — R	In	1 bit
clk	In	1 bit
clock — OUT_COUNT (	Output	N bits
counter		

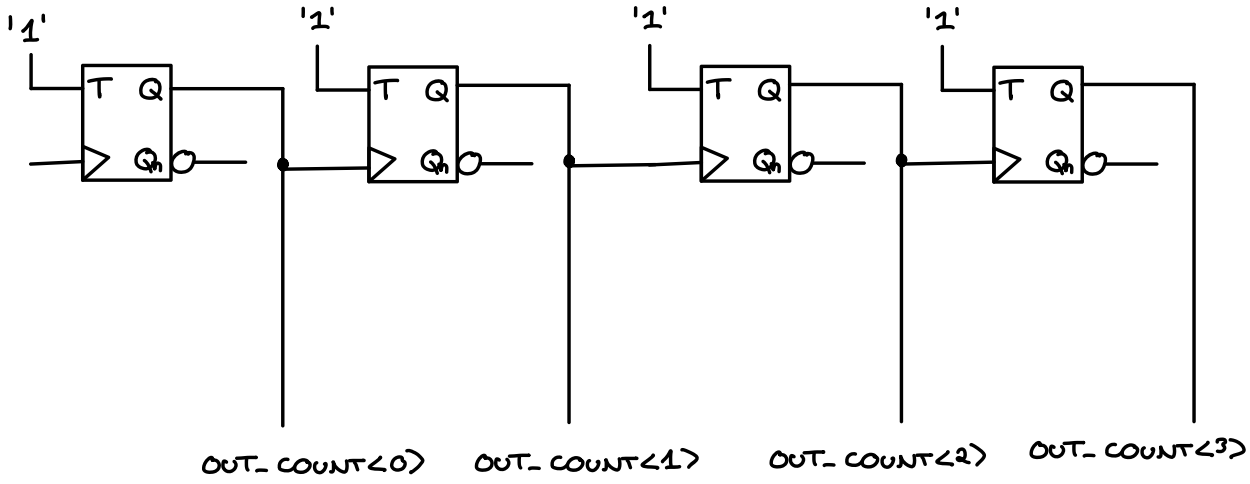
- Un Counter sincrono conta i fronti di salita del clock.

Digital Counter Flip-Flop-Level

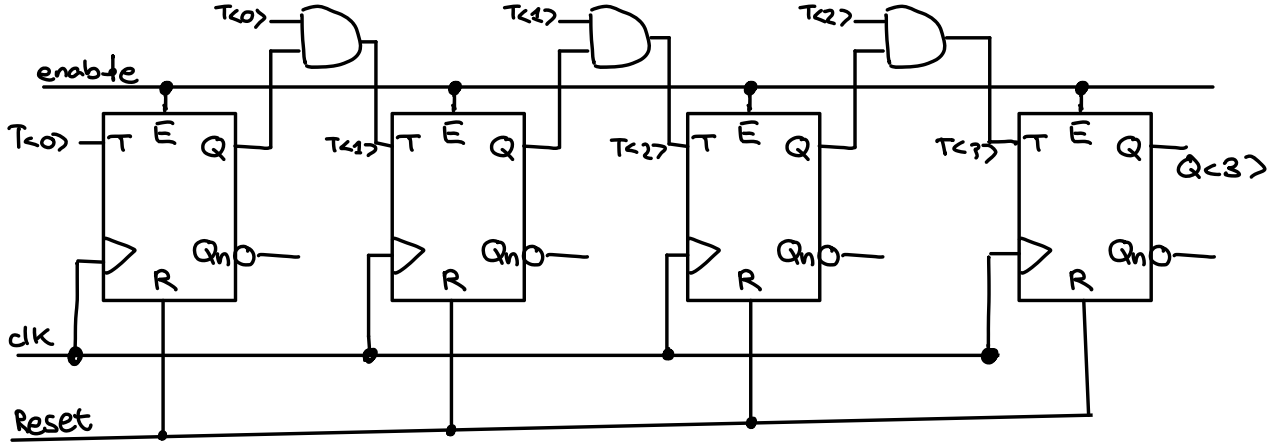
	STATE				TIME
T='1'	OUT_COUNT<3>	OUT_COUNT<2>	OUT_COUNT<1>	OUT_COUNT<0>	
T='1'	0	0	0	0	t_0
T='1'	0	0	0	1	t_0+1
T='1'	0	0	1	0	t_0+2
T='1'	0	0	1	1	t_0+3
T='1'	0	1	0	0	t_0+4
T='1'	0	1	0	1	t_0+5
T='1'	0	1	1	0	t_0+6
T='1'	0	1	1	1	t_0+7
T='1'	1	0	0	0	t_0+8
T='1'	1	0	0	1	t_0+9
T='1'	1	0	1	0	t_0+10
T='1'	1	0	1	1	t_0+11
T='1'	1	1	0	0	t_0+12
T='1'	1	1	0	1	t_0+13
T='1'	1	1	1	0	t_0+14
T='1'	1	1	1	1	t_0+15



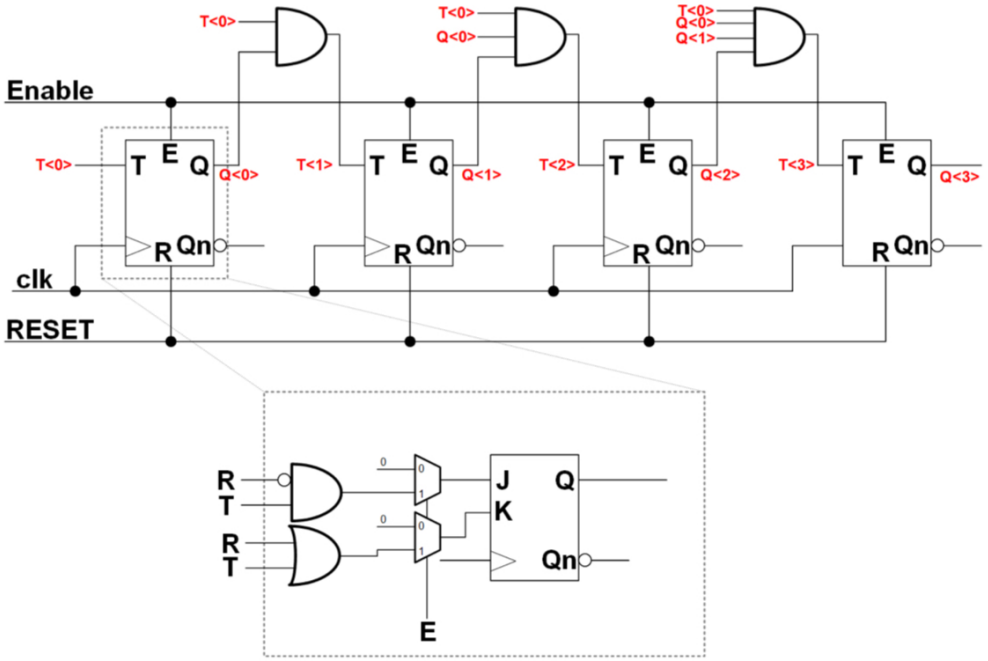
Input		Output	
$T(t_0-1)$	$Q(t_0-1)$	$Q(t0)$	$Q_n(t0)$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1



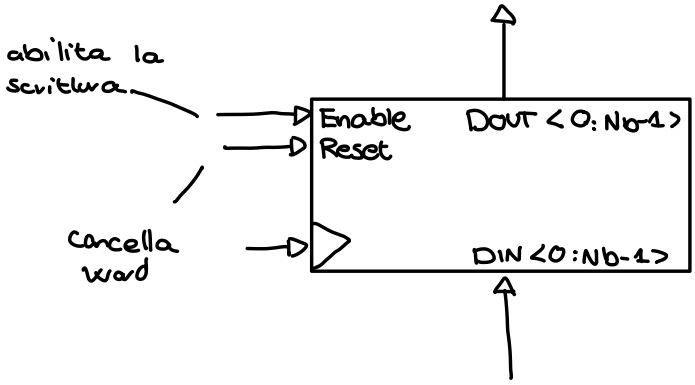
- Il segnale clock dell'i-esimo T-Flip-Flop è il Q uscita del precedente flip-flop.



	STATE				TIME
T='1'	OUT_COUNT<3>	OUT_COUNT<2>	OUT_COUNT<1>	OUT_COUNT<0>	
T='1'	0	0	0	0	t_0
T='1'	0	0	0	1	t_0+1
T='1'	0	0	1	0	t_0+2
T='1'	0	0	1	1	t_0+3
T='1'	0	1	0	0	t_0+4
T='1'	0	1	0	1	t_0+5
T='1'	0	1	1	0	t_0+6
T='1'	0	1	1	1	t_0+7
T='1'	1	0	0	0	t_0+8
T='1'	1	0	0	1	t_0+9
...	...	...	...	...	...
T='1'	1	1	1	1	t_0+15



Registers

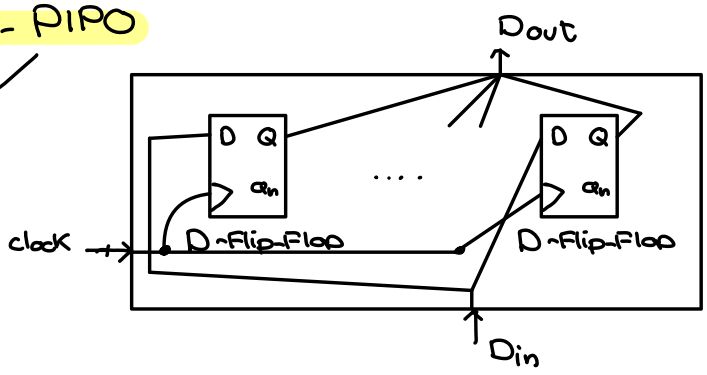


- Utilizzato per archiviare dati
- Hanno un dato di Input (la word da scrivere) e un Output (word scritta)

Registers - PIPO

Parallel-Input-Parallel-Output

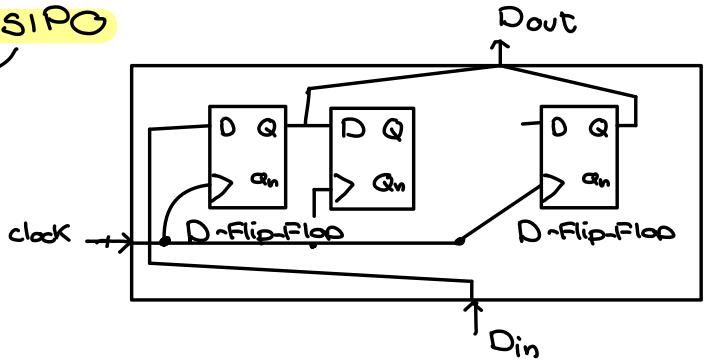
- Ho N bit, no N Flip-Flop



Registers - SIPO

Serial-Input-Parallel-Output

- L'output ha un delay dato dal clock



Registers - SISO

- Serial-Input-Serial-Output

Esercizio

Progettare un'interfaccia di comunicazione seriale/parallela (1→4) che riceve dati in ingresso (DATA_IN) in modo parallelo (REG_DATA).

Il circuito digitale che implementa la comunicazione ha le seguenti caratteristiche:

- si hanno a disposizione solo Flip Flop -D
- Enable e Reset

Disegnare un TIME-DIAGRAM

SOLUZIONE