

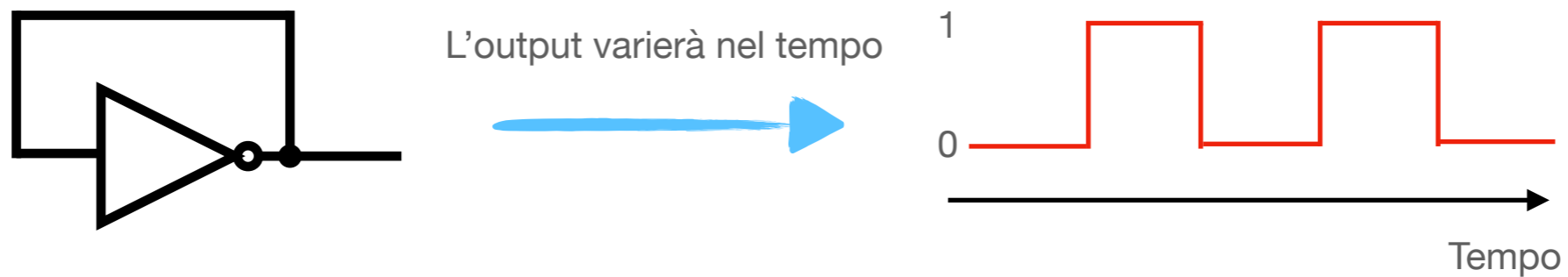
Programmazione e Architetture (Modulo B)

Lezione 4

Latches e Flip-flop

Rappresentare il tempo

Connettere l'output all'input



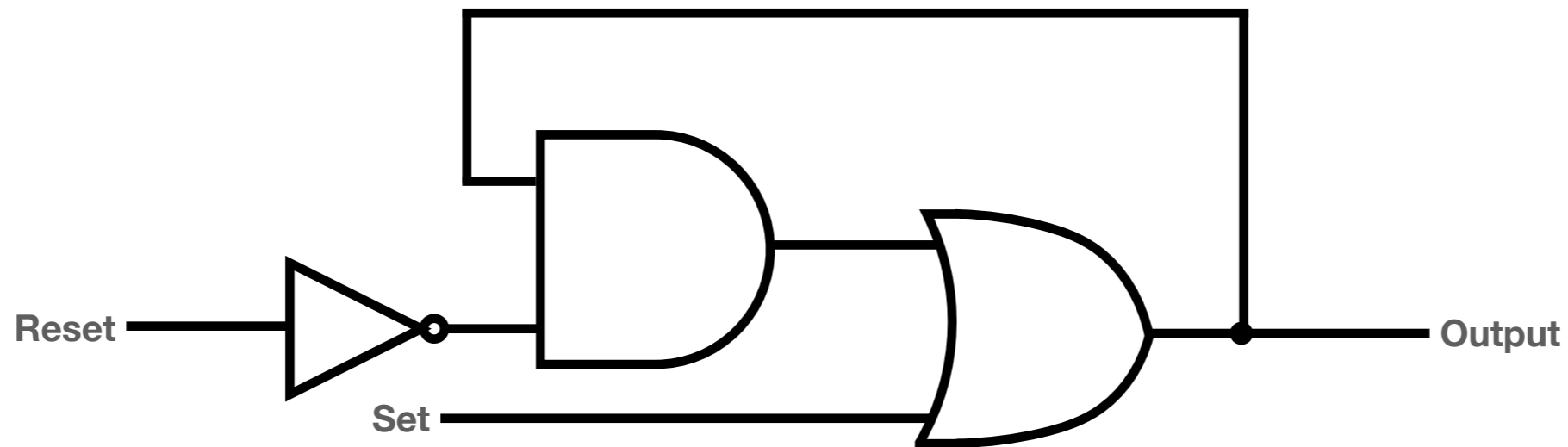
In realtà c'è un ritardo tra quando l'output viene prodotto in risposta a un dato input

Possiamo utilizzare questo clock per dare una nozione di tempo ai nostri circuiti logici.

Latches

Latch SR

Versione con AND e OR



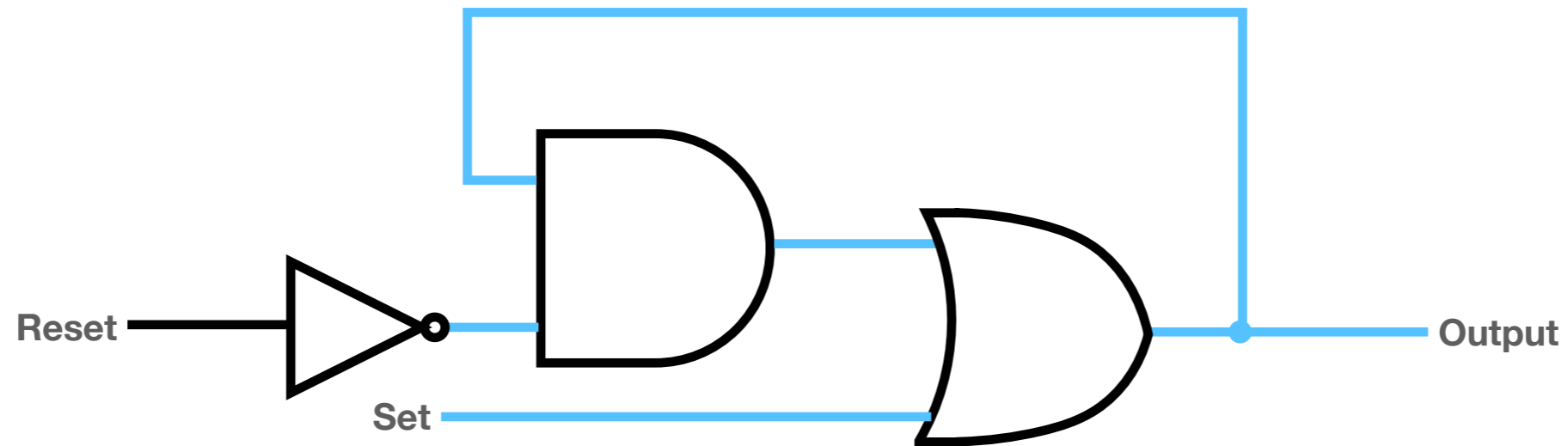
Questo semplice circuito ci permette di immagazzinare un bit di informazione

L'input **set** ci permette di avere di impostare l'output a uno

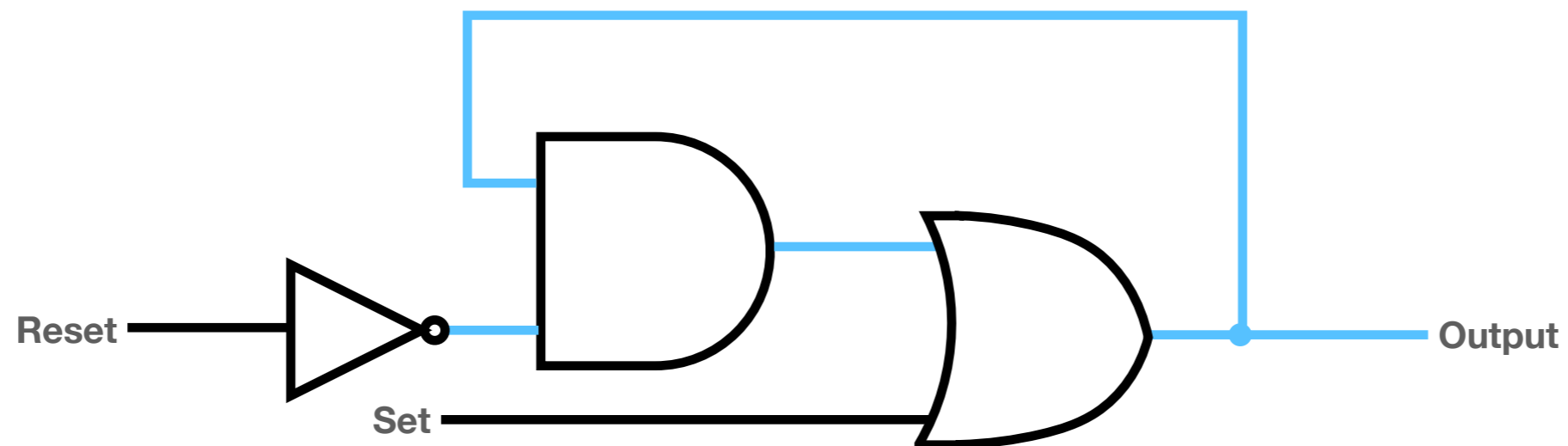
L'input **reset** ci permette di impostare l'output a zero

Latch SR: set

Versione con AND e OR



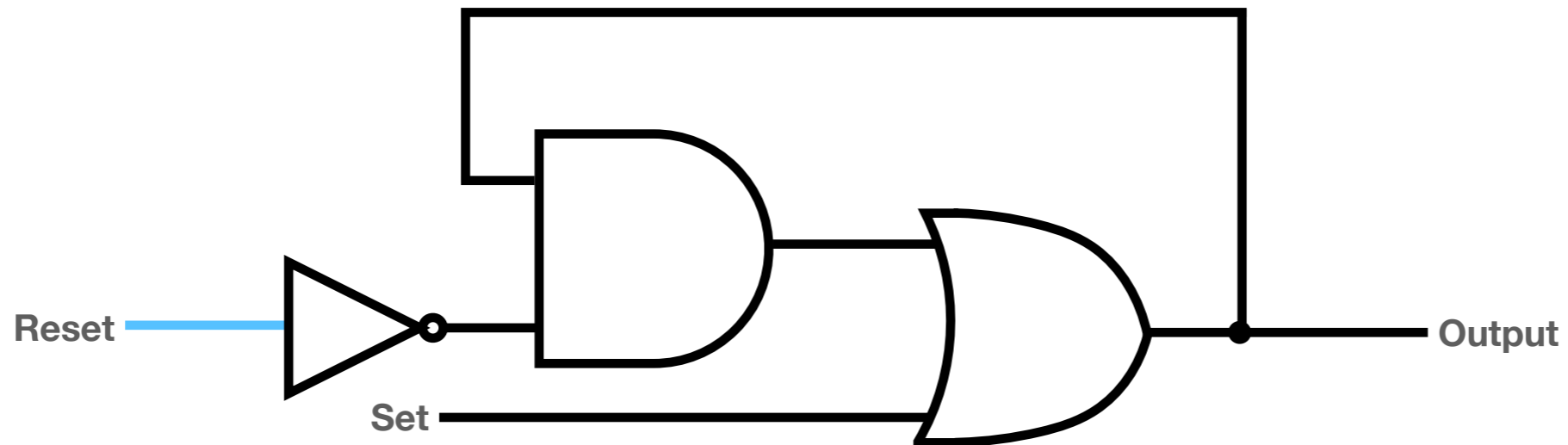
Se **reset** è zero e **set** è uno allora l'output viene impostato a uno



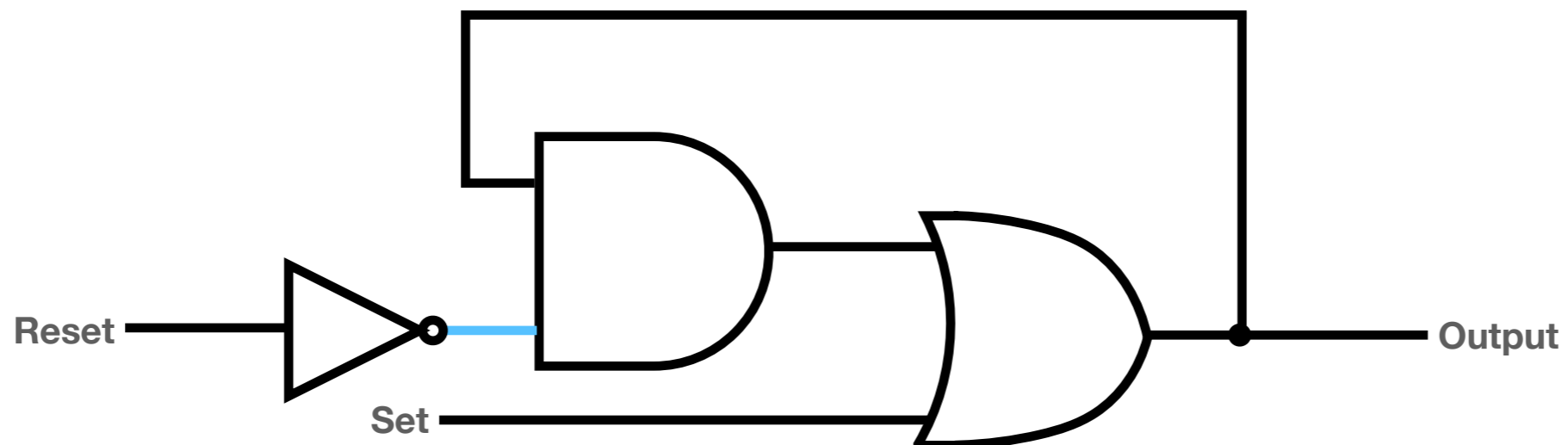
Ma l'output rimane uno anche quando **set** torna a zero!

Latch SR: reset

Versione con AND e OR



Quando **reset** è uno l'output diventa zero

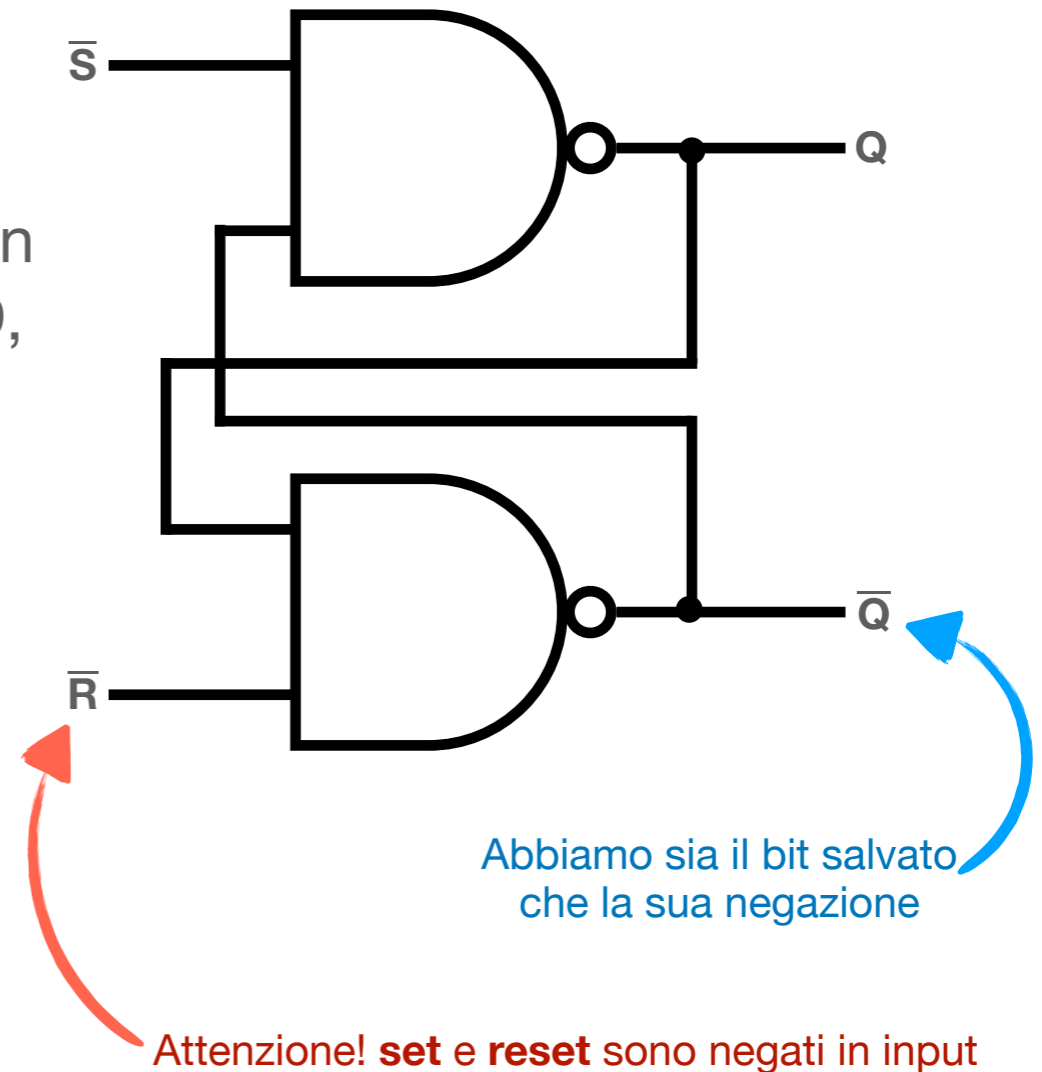


E rimane zero anche quando **reset** torna a zero

Latch SR

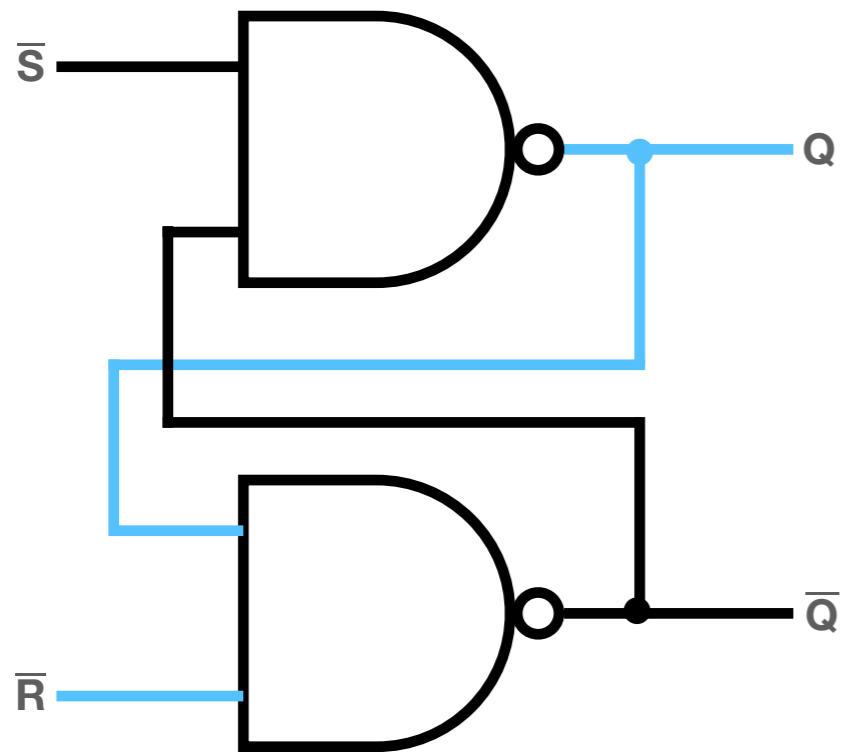
Versione con NAND

Quello che viene utilizzato non è solitamente un latch con **set** e **reset** realizzato con porte AND, OR e NOT, ma si possono utilizzare due porte NAND



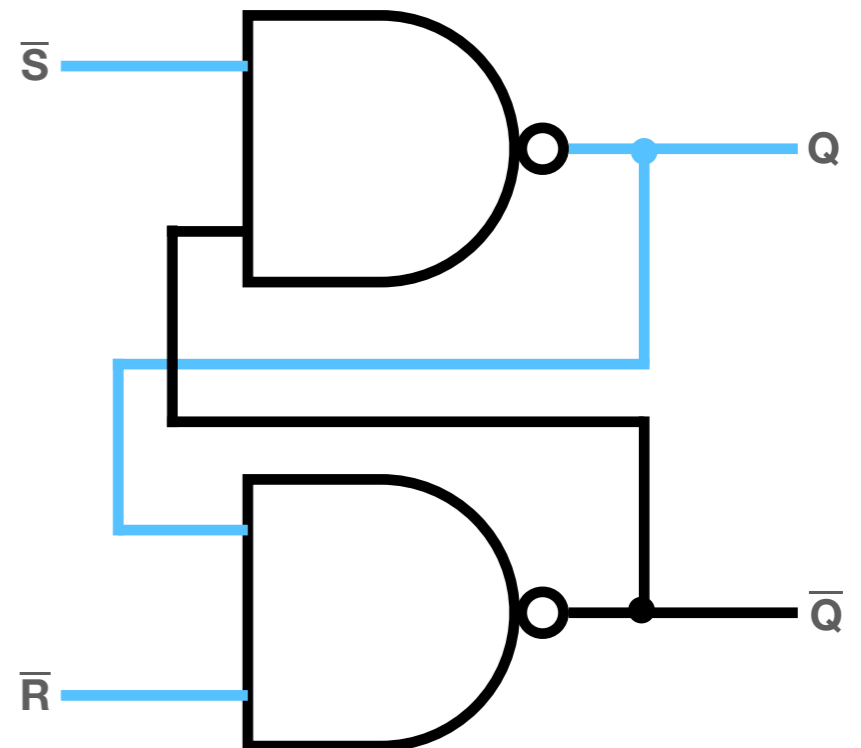
Latch SR: set

Versione con NAND



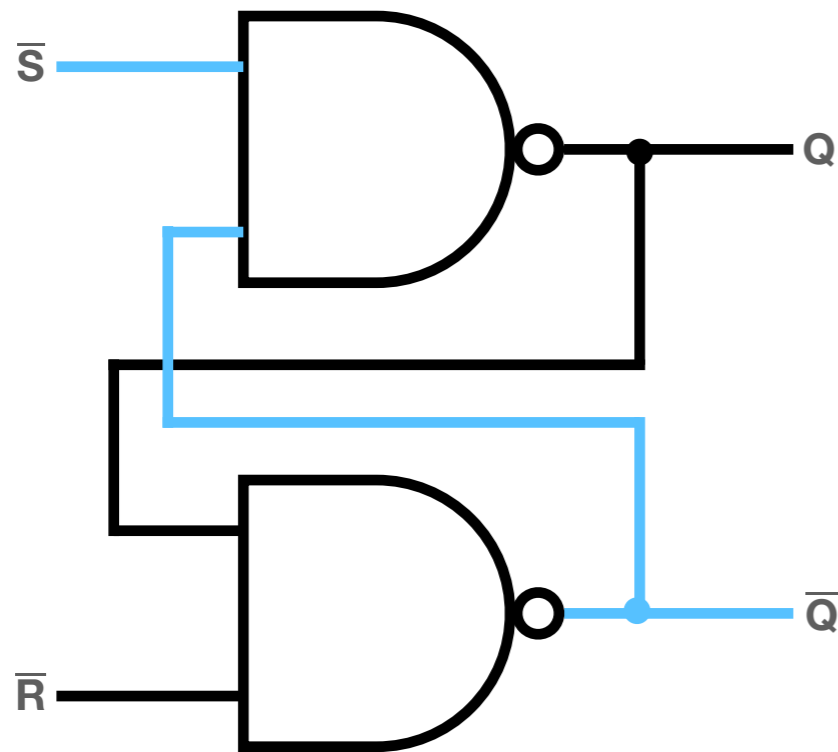
Se \bar{S} è zero (i.e., vogliamo impostare il bit salvato a uno), allora l'output diventa uno (e il suo negato zero)

Il bit di output rimane uno anche quando \bar{S} torna a uno



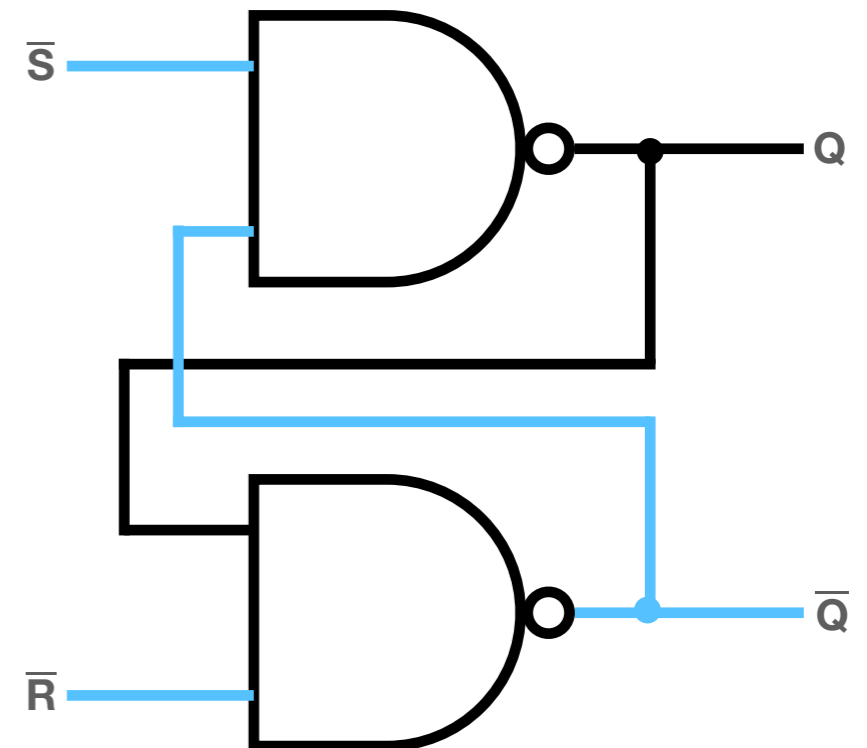
Latch SR: reset

Versione con NAND



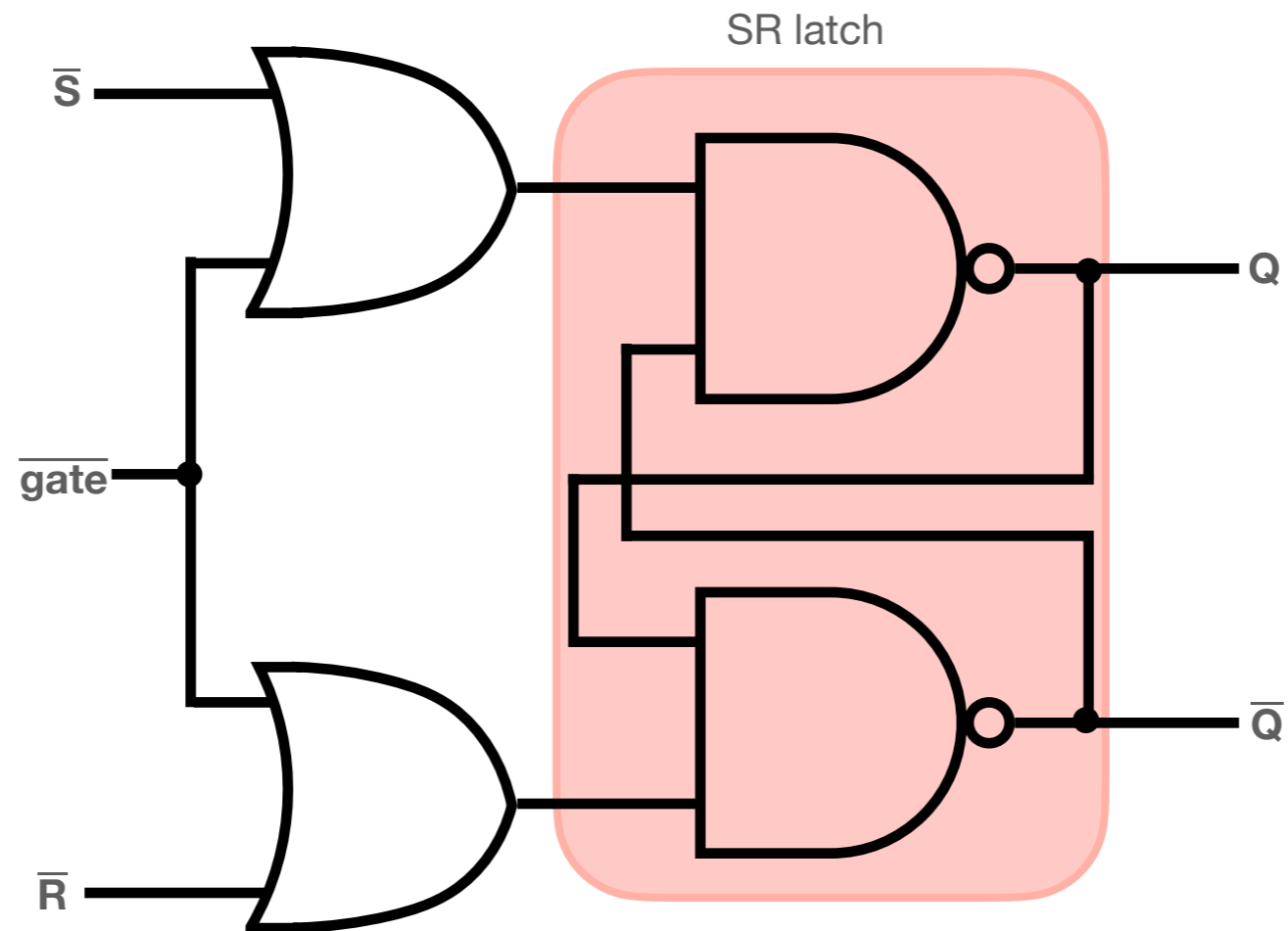
Se \bar{R} è zero (i.e., vogliamo impostare il bit salvato a zero), allora l'output diventa zero (e il suo negato uno)

Il bit di output rimane zero anche quando \bar{R} torna a uno



Gated SR Latch

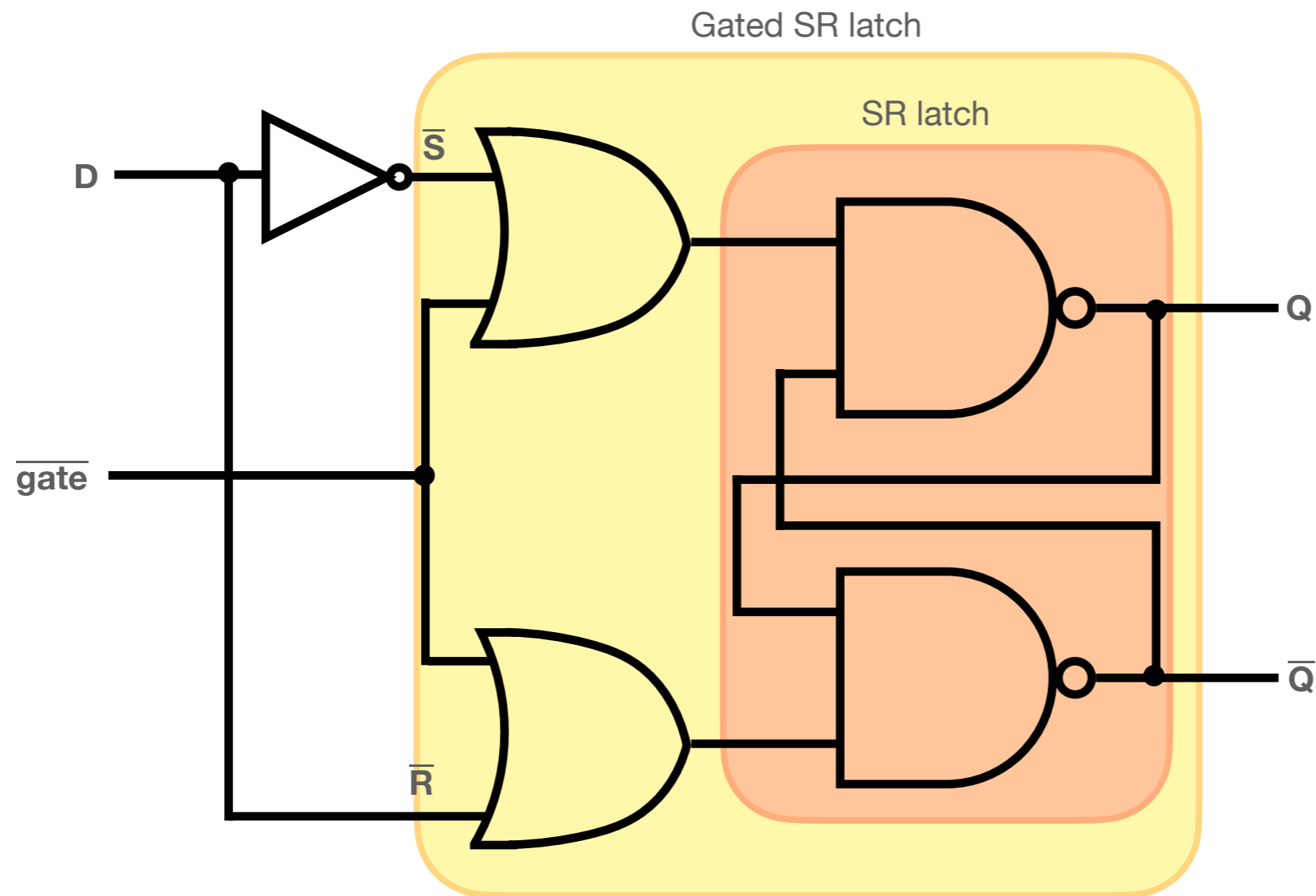
Controllare quando avviene la modifica



L'aggiunta di due porte OR fa in modo che la modifica del valore salvato sia possibile solo quando $\overline{\text{gate}}$ è a zero

Gated D Latch

Singolo input



Collegando entrambi gli input in modo che esattamente uno solo sia negato abbiamo che quando $\overline{\text{gate}}$ è a zero viene salvato nel gated D latch il valore attualmente nell'input **D**

Flip-Flop

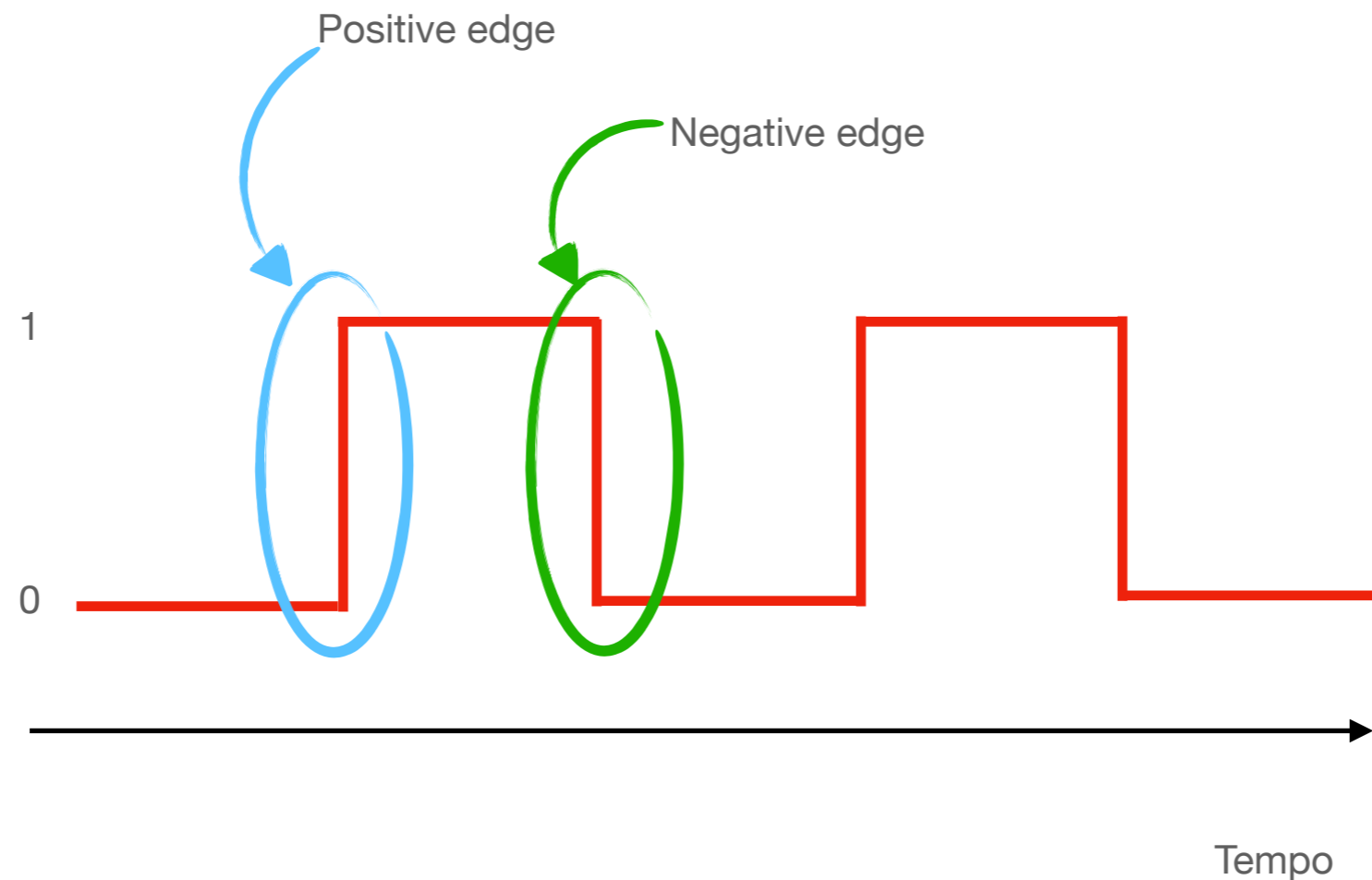
Edge-triggered latches

I flip-flop

- I D latches salvano il valore presente nell'input **D** quando il valore di $\overline{\text{gate}}$ è zero
- Se il valore di **D** cambia mentre lo stiamo salvando cambia anche quello che salviamo
- Però vorremmo minimizzare il “tempo di lettura”
- Per questo utilizzeremo il concetto di “edge triggered latches”

Positive e Negative Edges

Possiamo limitare la lettura a quando il valore di var varia da zero a uno (positive edge-triggered) o da uno a zero (negative edge-triggered)



D Flip-flop

Queste due parti servono a fare il modo che l'SR latch dopo di loro si attivi solamente quando il valore in **clock** passa da 0 a 1

clock

D

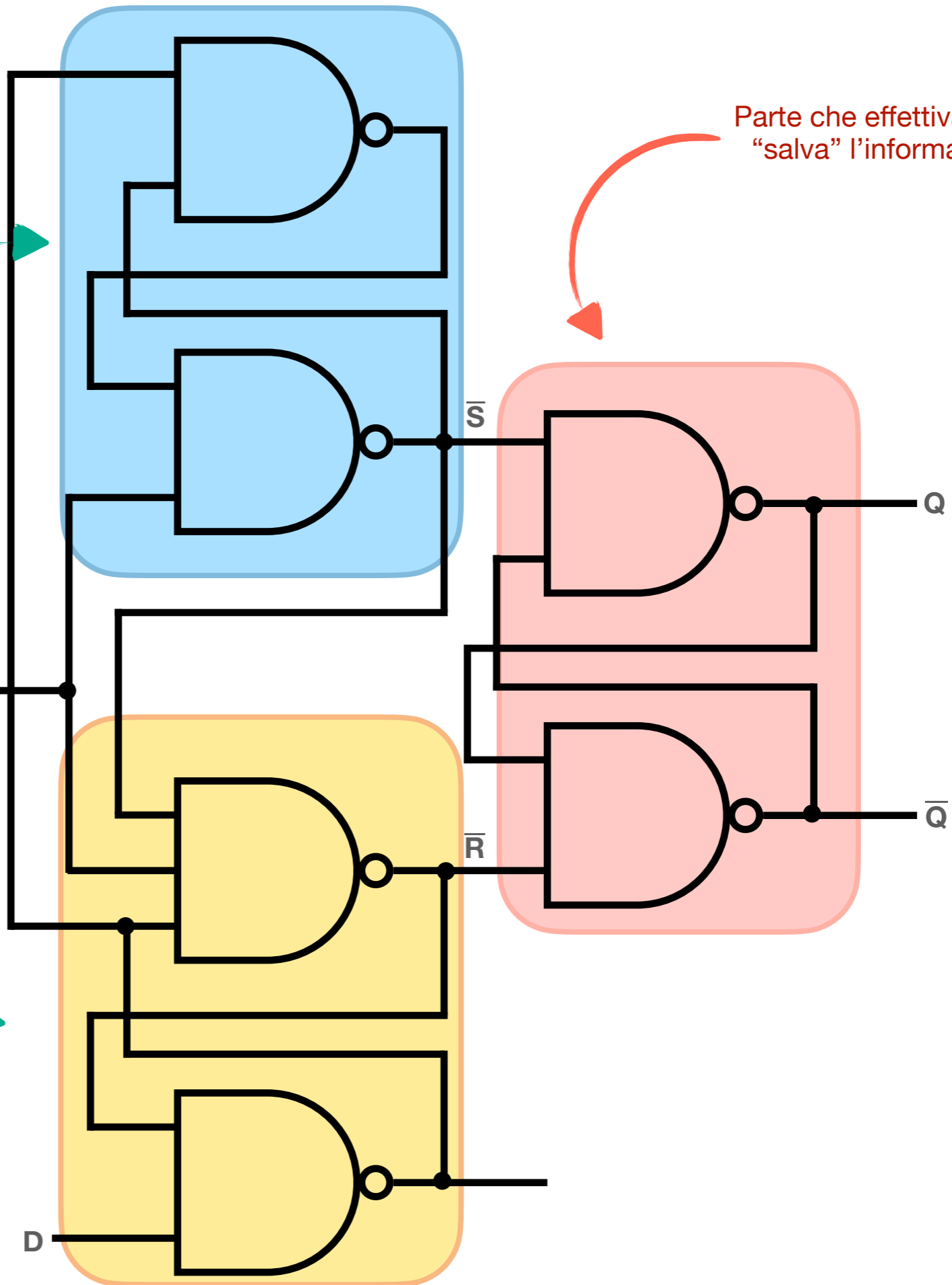
\bar{S}

\bar{R}

Q

\bar{Q}

Parte che effettivamente "salva" l'informazione



D Flip-flop

Possiamo ora “impacchettare” il D flip-flop e usarlo come “blocco” per costruire le nostre memorie

