

# Esperienza 12: Flip-Flop e contatori

Gruppo BN

Federico Belliardo, Lisa Bedini, Marco Costa

April 2, 2017

## 1 Flip-Flop D-Latch

Si è realizzato un circuito flip-flop di tipo D-Latch, come mostrato in figura 1 utilizzando le porte NAND di due integrati. L'ingresso D che corrisponde al dato da memorizzare è stato collegato all'impulsatore realizzato con Arduino Nano (in particolare a Y1 o a Y2?), e l'Enable è collegato alla tensione positiva attraverso uno switch manuale (inserire resistenza di pull-up?). La tensione di lavoro durante tutta l'esperienza è stata fissata a:  $V_{CC} =$ .

In figura ?? si vede come il segnale  $Q(t)$  in uscita dal flip-flop segua l'ingresso quando enable è alto riproducendo la tabella 1.

EN	D	S	R	Q
1	1	0	1	1
1	0	1	0	0
0	1	1	1	Hold
0	0	1	1	Hold

Table 1: Tabella degli stati per un Flip-Flop NAND.

Commutando manualmente lo switch e impostando dunque  $EN = 0$  il flip flop rimane congelato nello stato in cui si trovava prima della commutazione. Perché il valore che il flip-flop memorizza sia deterministico è necessario che la commutazione dello switch non avvenga durante gli hold-time e setup-time del latch. In figure ?? e ?? si vede il flip-flop congelato nei due stati. Quando il bit di enable è disattivato entrambe le uscite dei NAND del primo livello sono a 1 pertanto il latch è nello stato di hold.

Essendo il latch costruito con delle porte NAND ho una situazione di instabilità quando gli ingressi delle porte sul secondo livello sono entrambe a 0. Questo può succedere solo se gli ingressi di tutte le porte sul primo livello sono 1. IL NOT evita questa situazione.

L'enable è attivo alto. Cioè quando  $enable = 0$  ho permanenza dello stato, infatti gli ingressi al secondo livello dei NAND sono sicuramente a 1. mentre posso avere evoluzione dello stato se il bit  $enable = 1$ .

Si sono misurati i tempi di ritardo sulla salita e discesa del flip-flop (quando è abilitato) essi sono risultati asimmetrici. Le misure sono riportate nella tabella 2.

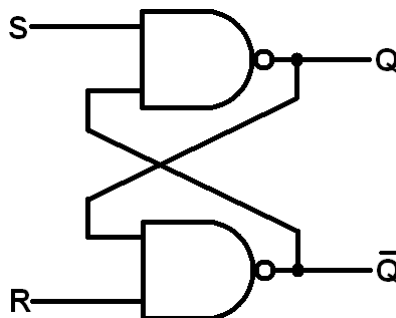


Figure 1: R-S NAND Latch.

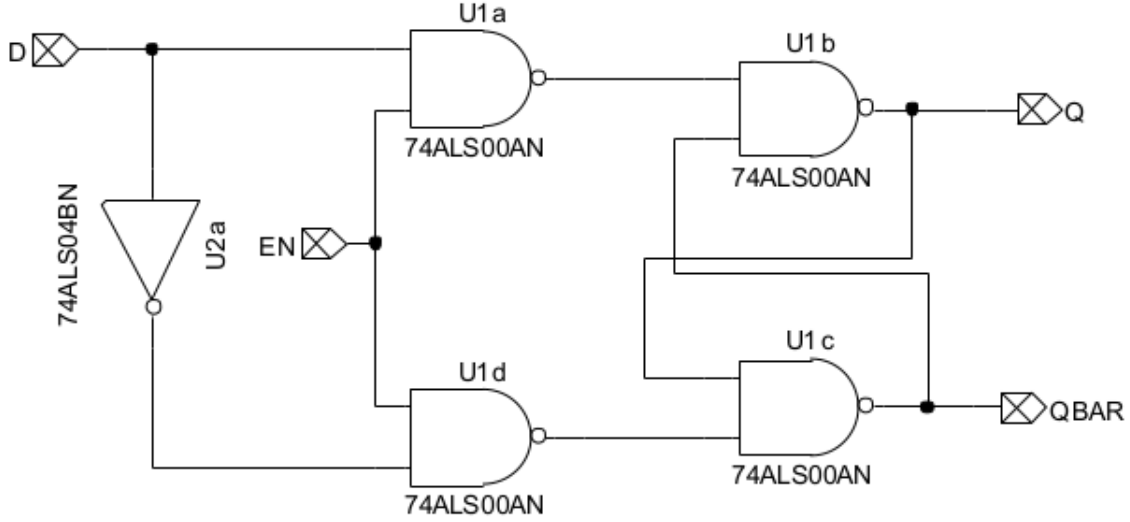


Figure 2: Circuito D-Latch NAND.

$t_{LH}$ (ns)	$t_{HL}$ (ns)
tempo 1	tempo 2

Table 2: Misure dei tempi di ritardo in salita e discesa per il Flip-Flop.

## 2 Divisori di frequenza

Si è realizzato un contatore a 4 bit come in figura 3 connettendo l'uscita Q del primo JK (FF1) al clock del secondo JK (FF2), i restanti JK sono già interconnessi all'interno del package. Si vogliono utilizzare tutti i JK in modalità Toggle, cioè in modo che oscillino tra lo stato 0 e 1 alla transizione basso-alto del clock (per costruzione). Per fare ciò è necessario che l'ingresso J di ognuno dei flip-flop sia impostato a 1, questa configurazione è realizzata ponendo  $R_0$  a terra e  $R_1$  flottante (per il momento).

Infatti la configurazione toggle del JK si ottiene se  $J = 1$  e  $K = 1$  dunque con K flottante questo è sempre uguale a 1, se uno degli ingressi del NAND è forzato a 0 la sua uscita sarà sempre 1 e sono quindi nella richiesta per i JK.

Poiché la transizione del valore di uscita Q di ognuno degli FF avviene con una frequenza dimezzata rispetto a quella del clock, le varie uscite  $Q_i(t)$  oscillano con frequenze  $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}$  in quanto l'uscita  $Q_i$  di un FF è il clock del successivo.

Ogni uscita  $Q_i$  è stata collegata a terra attraverso un LED e una resistenza (per limitare la corrente sul led). Questo e l'accorgimento di impostare un clock per il FF1 di 1 Hz (generato con Arduino) rende osservabili ad occhio nudo le transizioni.

Interpretando il bit più a sinistra come bit meno significativo (1 per LED acceso e 0 per LED spento) si ottiene la rappresentazione fisica dei numeri da 0 a 15 in binario.

Si è inviato in ingresso (sempre con Arduino) un segnale di frequenza  $f = kHz$ , si sono misurate le frequenze dei segnali  $Q_i$  e i tempi di ritardo per la salita e la discesa rispetto al clock, definiti come differenza temporale i punti in cui i due segnali raggiungono la metà del rispettivo valore massimo.

•	$T(ms)$	$f(kHz)$	$t_{HL}$	$t_{LH}$
$Q_2$	•	•	•	•
$Q_2$	•	•	•	•
$Q_3$	•	•	•	•
$Q_4$	•	•	•	•

Table 3: Misure di frequenza e tempi di propagazione per il divisore.

I tempi di ritardo misurati sono tempi di propagazione attraverso la rete dei flip-flop e dalle misure eseguite aumentano linearmente con il numero di porte del circuito. Il tempo di propagazione da *low* a *high* è sistematicamente più alto di circa  $\Delta t = ...ns$ , come si può vedere nella tabella 3.



Il contatore è edge-triggered sul fronte di discesa del clock, mentre il D-Latch è attivo se il clock è alto, dunque perchè il reset sia sincrono è necessario usare come clock del D-Latch il clock negato.

Si è realizzato il circuito in figura 5 e inviando un clock a bassa frequenza si osservata (e riportata in seguito) la sequenza generata (che come si vede è completa). Si sono provate tutte le possibili combinazioni di "tap" (ingressi  $Q_i$  dello XOR) e si è visto che sono due danno sequenze complete:  $Q_2, Q_3$  e  $Q_0, Q_3$ .

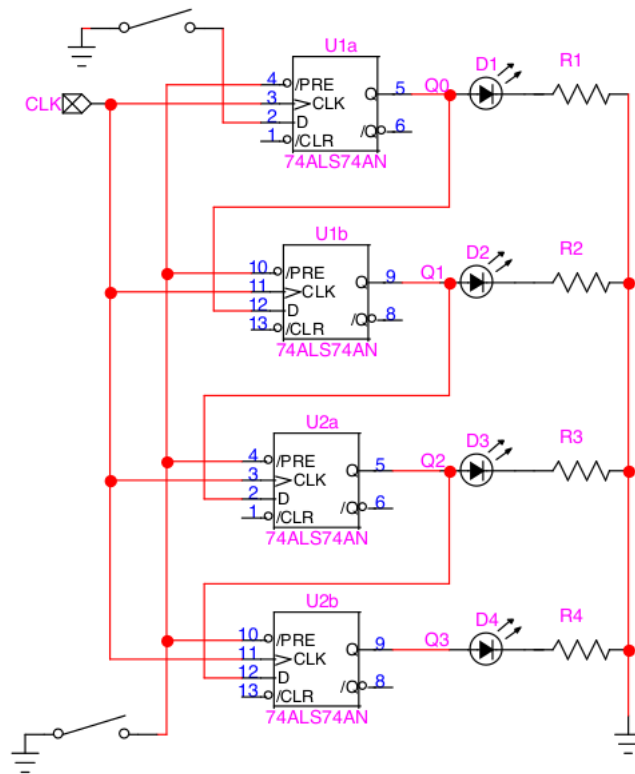


Figure 4: Shift register.

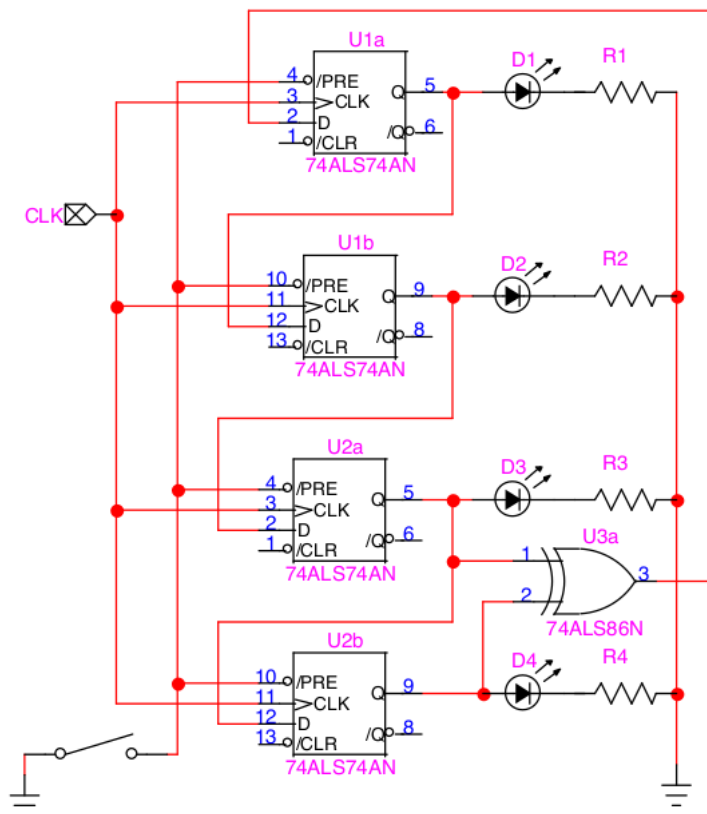


Figure 5: Generatore di numeri pseudo-casuali.