

Scheda di laboratorio n.09

Generatore di sequenze di bit pseudocasuali

In questa esperienza completeremo un generatore di numeri pseudocasuali grazie all'implementazione di un registro di scorrimento e usando la porta XOR costruita durante l'ultima esperienza. Prima di procedere con l'assemblaggio del circuito finale, prenderemo confidenza con i registri di memoria, che useremo per realizzare dei divisori di frequenza.

1 Flip-flop e divisori di frequenza

I *flip-flop*¹ possono essere usati come divisori di frequenza, ossia dei circuiti che, ricevendo in ingresso un segnale logico con periodo di ripetizione $T = 1/f$, sono in grado di generare in uscita un segnale con un periodo che è un multiplo di T . Nei prossimi esercizi utilizzeremo in particolare dei *flip-flop* di tipo D.

Task 1 Scaricare il datasheet dell'integrato CD4013 individuare il significato di ognuno dei pin dell'integrato, che contiene *due* D *flip-flop*. Ricordate che nessun pin di ingresso del *flip-flop* va lasciato flottante, quindi assicuratevi di capire come configurarlo a seconda di che cosa volete ottenere. Riportare sul logbook anche la piedinatura dell'integrato.

Un primo esempio di divisore è visibile sul lato sinistro della figura 1.1 ed è stato già illustrato a lezione: dato un ingresso a frequenza f , fornisce una uscita a frequenza $f/2$.

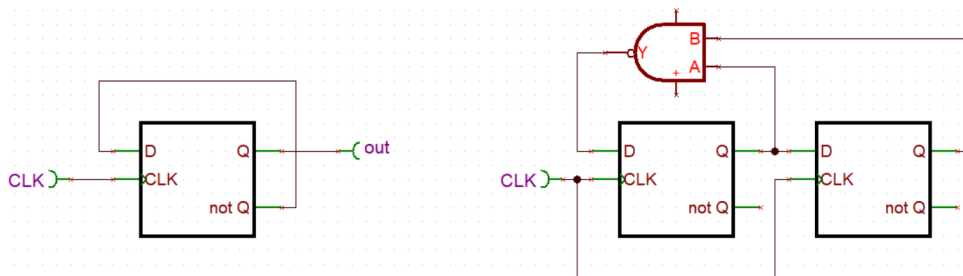


Figura 1.1: Divisori di frequenza realizzati con D *flip-flop*.

Task 2 Realizzare il divisore di frequenza per due e collegare l'ingresso CLK ad un segnale di *clock* realizzato generando un'onda quadra sul canale W1, compresa tra 0 e 5 V, di frequenza a piacere. Misurare quindi sia il *clock* che l'uscita del circuito tramite Ch1 e Ch2, dimostrando la corretta operazione del divisore. Ricordiamo che per far funzionare correttamente il circuito vanno opportunamente configurati gli ingressi SET e RESET del *flip-flop*. La configurazione va annotata sul logbook.

Si consideri ora il circuito riportato sulla destra della figura 1.1. Dato che è composto da due *flip-flop*, il circuito può trovarsi in quattro configurazioni distinte: 00, 01, 10 e 11.

Task 3 Ricostruire tabella di verità e il diagramma di stato del circuito, descrivendo quindi quale è l'evoluzione temporale di ognuna delle quattro configurazioni citate, dopo un ciclo di *clock*. Tracciare quindi il diagramma temporale atteso includendo ognuno dei segnali del circuito. Ci aspettiamo che il circuito si comporti come un divisore di frequenza? Si tratta di un circuito sincrono o asincrono?

¹ che, ricordiamo, sono per definizione *edge-triggered* altrimenti li chiamiamo em latch.

Task 4 Realizzare il circuito usando uno dei NAND dell'integrato CD4011 e verificare sperimentalmente la correttezza del diagramma temporale ricavato nel precedente esercizio. Riassumere i risultati in una misura `Divisori.pdf` in cui mostrare l'evoluzione temporale del *clock* e dell'uscita dei due divisori di frequenza. Si noti che non è affatto detto che l'onda finale risultante abbia un *duty cycle* del 50%.

2 Registri a scorrimento

Il prossimo ingrediente dell'esperienza saranno i registri a scorrimento che, come illustrato a lezione, sono tipicamente costituiti da una sequenza di flip-flop di tipo D che sono in grado di memorizzare il valore logico presente al pin di ingresso durante il fronte di salita del segnale di *clock*. Quando vari flip-flop di questo tipo sono connessi in serie, ogni fronte di salita del *clock* farà scorrere i bit di una posizione lungo la sequenza di celle di memoria. Nel caso specifico, faremo pratica con il chip MC14557, che contiene un registro lineare a scorrimento con un numero configurabile di bit e del chip CD4013 che contiene due *flip-flop* di tipo D.

Task 5 Scaricare il datasheet di MC14557 e individuare il significato di ognuno dei pin dell'integrato. Riportare sul logbook anche la piedinatura dell'integrato. Studiare il diagramma logico di MC14557 riportato a pagina 2 del datasheet e cercate di capire come è possibile che questo chip realizzi un registro di lunghezza variabile. Spiegare il funzionamento di massima del dispositivo in al più due frasi da includere nel logbook.

Per controllare il funzionamento del registro verificheremo se un fronte d'onda logico scorre, come dovrebbe, dall'ingresso fino all'uscita dopo un numero sufficiente di cicli di *clock*. Lo scorrimento del registro sarà controllato dal segnale CLK generato da W1. Il secondo canale W2 verrà invece usato per generare un'altra onda quadra *sufficientemente lenta rispetto al clock* (lasciamo scegliere a voi quanto), che verrà iniettata all'input A del registro e che ci aspettiamo di vedere comparire all'uscita Q dopo un certo tempo Δt (figura 2.1). Raccomandiamo una attenta lettura del datasheet dell'integrato, per decidere quali valori logici assegnare ai vari piedini di ingresso. Nel di incertezze, considerate con i bassi voltaggi utilizzati è sostanzialmente impossibile danneggiare i chip; fate quindi senza eccessivi timori i test del caso per capire come funziona i circuito che state montando.

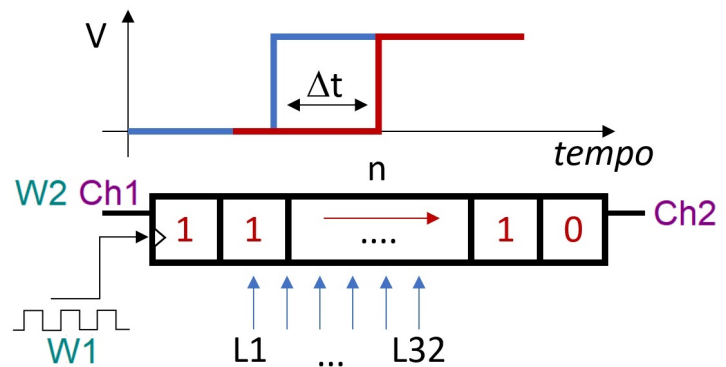


Figura 2.1: Ritardo di propagazione attraverso un registro a scorrimento di un fronte d'onda logico.

Task 6 Collegare il registro di scorrimento al Analog Discovery 2 come indicato in Fig.2.1 e misurare il tempo che impiega una onda quadra generata sul canale W2 a propagare attraverso gli n di bit del registro. Il segnale in input W2 deve essere mandato sull'ingresso A, dato che l'ingresso B sarà usato in seguito per altro. Il segnale CLK generatore da W1 va invece inviato all'ingresso chiamato CLK. Oltre a questo i seguenti pin non vanno lasciati flottanti e vanno impostati al valore logico corretto (datasheet!): RESET, \overline{CE} , A/B e più ovviamente tutti i vari selettori di lunghezza L1, L2, eccetera, che fissano n . Effettuare la misura quando il registro è configurato in modo da avere $n = 64, 48, 32$ e 16 bit e verificare la consistenza con il periodo del *clock* impostato.

Sincronia dei segnali. Nel punto precedente si sarete certamente scontrati con la difficoltà legata al fatto che, in generale, non è chiaro quale sia l'allineamento fra i fronti d'onda di W1 e W2. In effetti, in assenza di opportuna configurazione, nulla garantisce una generazione sincrona con fasi relative prevedibili. Facciamo notare due sintassi possibili con il modulo `tdwf`, che si appoggiano ad opportune funzioni nella SDK dello strumento. Si supponga che `wgen` sia una istanza opportunamente configurata del generatore di funzioni

[no sync]	W1 e W2 indipendenti	[sync]	W1 e W2 sincroni
<code>wgen.w1.start()</code>	# Avvia W1	<code>wgen.w1.sync()</code>	# impone W1 in sync con W2
<code>wgen.w2.start()</code>	# Avvia W2 (no sync)	<code>wgen.w2.start()</code>	# Avvia W1 e W2 in sync

Task 7 Collegare l'uscita Q di MC14557 all'ingresso D di uno dei due *flip-flop* contenuti in CD4013, e realizzare un registro a $(n + 1)$ bit. Collegare entrambe gli integrati al segnale CLK e verificare semplicemente se il segnale del DAC propaga fino all'uscita del flip-flop D, ossia che se A viene impostato ad un livello logico alto (basso) allora dopo qualche ciclo di *clock* l'uscita Q di CD4013 diventa alta (bassa); questo conferma il corretto montaggio. Anche qui, ricordiamo che è necessario collegare a GND o a 5V ognuno dei pin di configurazione del *flip-flop* utilizzato: lasciarli flottanti tipicamente porta a comportamenti poco prevedibili.

3 Generatore di numeri pseudocasuali

A questo possiamo chiudere il ciclo di feedback e configurare il nostro registro a scorrimento come un generatore di numeri pseudocasuali. A questo fine sono necessarie le seguenti operazioni:

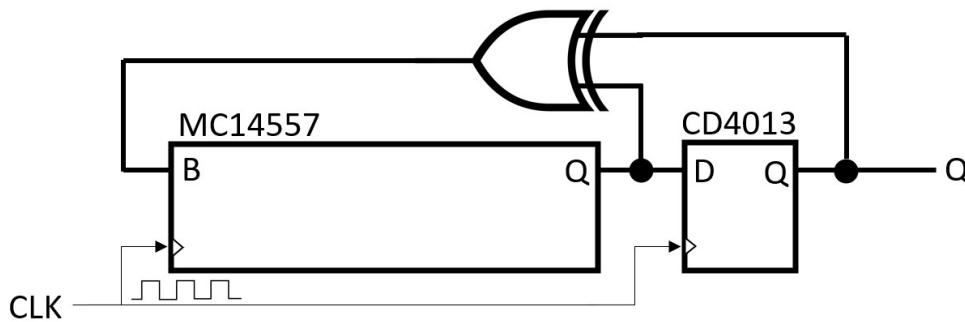


Figura 3.1: Circuito finale: generatore di bit pseudocasuale.

- Collegare l'uscita Q di MC14557 e l'uscita Q del flip-flop collegato di CD4013 agli ingressi della porta XOR realizzata in precedenza;
- Collegare l'uscita dello XOR all'ingresso B al registro a scorrimento.
- Connettere il pin A/B a un cavo che si possa riconfigurare in maniera comoda, dato che ci potrebbe servire cambiare il suo livello logico.
- Collegare la linea CLK all'uscita W1, che useremo per generare un segnale di clock con frequenza a vostra discrezione.
- Per l'occasione consigliamo di collegare sull'output Q finale un LED in maniera simile a quanto fatto con il *latch SR*.

Nel seguito indicheremo una serie di esercizi dove n sarà sempre il numero di bit del chip MC14557, e quindi il registro inclusivo del flip-flop D avrà $n + 1$ bit. Suggeriamo di prestare la attenzione a come un dato n venga codificato dai livelli dei vari ingressi L1, L2, eccetera.

Homework 1 Calcolare (nel modo che preferite) le sequenze di bit che sono generate dal circuito per i vari n che vanno almeno da 1 a 5, partendo da una configurazione binaria $1 \cdots 1$. Plottare quindi la forma d'onda attesa nei vari casi e annotare sul logbook il periodo di ripetizione.

Task 8 Realizzare il circuito di Fig.3.1 e dimostrare sperimentalmente il funzionamento del generatore configurando inizialmente MC14557 su $n=3$. Misurare la sequenza di bit pseudocasuale e mostrare che coincide con quella teorica (che si può rapidamente calcolare perfino a mano). In seguito, ripetere l'operazione per tutti gli n mancanti che vanno da 1 a 5.

Mentre fino a questo punto le misure e la verifica della sequenza potevano essere effettuate sostanzialmente a mano, al crescere di n questo tipo di approccio non è pratico. Nel prossimo esercizio vi verrà chiesto di spingere questa misura a n più alti possibile e il suggerimento è di sviluppare delle tecniche per analizzare in maniera più possibile automatica il dato misurato².

²e magari anche per misurarlo: potete modificare il circuito e il software a vostro piacimento

Task 9 Costruite un codice MATLAB che, partendo dall'acquisizione analogica dell'output del generatore di numeri pseudocasuali, estragga una lista di valori numerici corrispondenti ai bit.

Task 10 La sequenza di bit generata dal circuito non è davvero casuale, ma bensì deterministica. In particolare, dato che il registro può trovarsi solo un numero finito di configurazioni che determinano in maniera univoca l'evoluzione nel tempo, la sequenza pseudocasuale deve avere sempre una periodicità finita. Variate n e cercate di dimostrare il più lungo periodo di ripetizione possibile con il circuito montato. Quanto vale? Vi chiediamo tre tipi di valutazione:

- **Teorica.** Non è richiesta una dimostrazione matematica formale, basatevi pure su qualsiasi fonte (fidata) riusciate a trovare sul comportamento di questa classe di circuiti;
- **Numerica,** ossia calcolando la sequenza numericamente per un dato n , sostanzialmente estendendo il lavoro svolto fino a $n = 5$;
- **Sperimentale,** ossia facendo una misura sul circuito montato e dimostrando che il dato ha una certa periodicità.

Dato il livello di complessità non del tutto trascurabile del circuito, lasciamo qualche indicazione/consiglio per capire quale può essere una buona procedura per avviarlo e gestirlo.

Procedura suggerita di innesco. La condizione di accensione per il registro di shift usato è quella di RESET, ossia con tutti i bit a valore zero. Sfortunatamente, questa configurazione non ha nessuna evoluzione temporale nel registro di shift montato, come è facile verificare a mano guardando le operazioni svolte dallo XOR. Per avviare il ciclo pseudocasuale è necessario poter configurare il registro su un valore diverso. Nella seconda parte dell'esperienza raggiungeremo un grado di controllo superiore del circuito, ma per ora possiamo impostare tutti i bit a un valore logico *vero* con la seguente sequenza di operazioni:

- Configurare il selettore A/B in maniera da attivare l'ingresso A, che va collegato ad un valore logico alto. Questo dovrebbe rapidamente caricare un valore 1 in tutti gli $n + 1$ bit del registro;
- Riconfigurare il selettore A/B in maniera da ristabilire il ciclo di retroazione tramite l'ingresso B;

Nel caso di problemi (ma non dovrebbe succedere), è possibile anche bloccare il *clock* usando ECLK prima del secondo passo e fare ripartire il circuito solo quando il feedback è stato rimesso chiaramente in funzione. Una volta eseguite queste operazioni il generatore dovrebbe iniziare a funzionare a ciclo continuo. L'operazione va ripetuta solo nel caso in cui venga tolta l'alimentazione o, a volte, nel caso in cui riconfigurate il valore di n (è possibile farlo "a caldo", ossia mentre il circuito funziona... non è pericoloso, al più da dei risultati poco prevedibili).

Frequenza di operazione del circuito e feedback visivo. Sebbene non sia richiesto, consigliamo di realizzare un feedback visivo immediato dello stato del generatore di numeri casuali collegando l'uscita Q del bit finale su CD4013 anche^a ad una resistenza e quindi ad un LED che va a GND, in maniera del tutto simile a quanto fatto con il *latch* SR. Se/quando volete vedere le singole transizioni è consigliato abbassare la frequenza del segnale CLK.

^a "anche" nel senso che ovviamente non dovete scollegarlo dall'ingresso dello XOR altrimenti il circuito smette di funzionare!