



Esercizi

Logica Digitale

Alessandro A. Nacci
alessandro.nacci@polimi.it

ACSO
2014/2014



Esercizio 1 - Prima Parte

Si vuole realizzare un circuito combinatorio a quattro ingressi (a, b, c, d) e un'uscita F, caratterizzato dai *mintermini* seguenti (0, 1, 2, 5, 10, 11, 15).

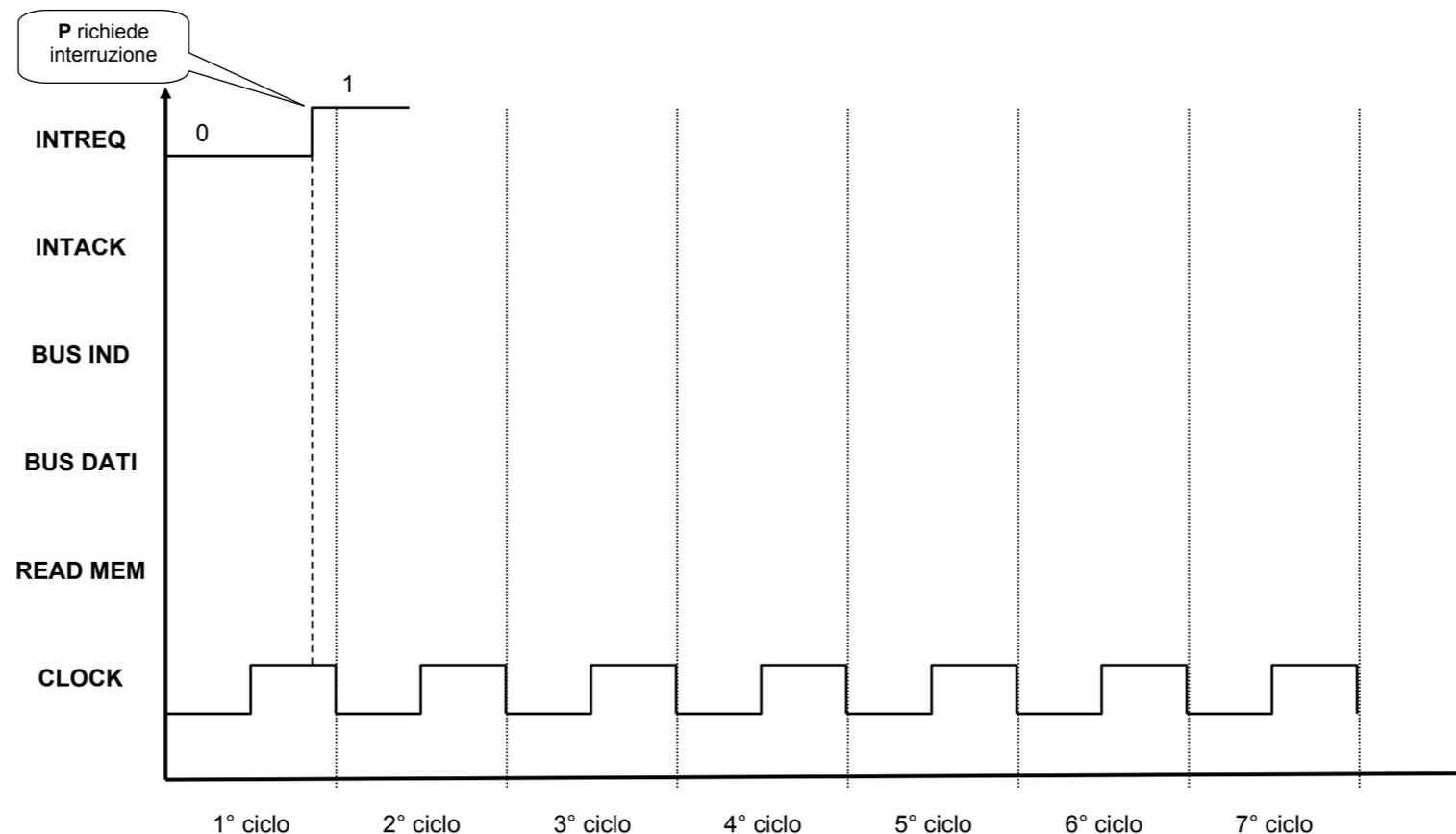
- A. Si scriva la prima forma canonica di F:
- B. Si disegni la mappa di Karnaugh della funzione evidenziando tutti gli implicant primari
- C. Si sintetizzi la funzione tramite il metodo delle mappe di Karnaugh e si scriva la forma minima come somma di prodotti (SOP). Qualora esistano più forme minime le si indichino tutte (il numero di righe date sotto non è significativo)
- D. Si disegni una rete combinatoria che realizza la funzione calcolata al punto precedente, utilizzando solamente porte a due ingressi.
- E. Si calcoli il ritardo della rete disegnata al punto precedente supponendo che ogni porta AND a due ingressi abbia ritardo di 7 ns e che ogni porta OR a due ingressi abbia ritardo pari a 5 ns (non si consideri il ritardo delle porte NOT applicate agli ingressi primari).

Esercizio 1 - Seconda Parte

Si chiede di completare il diagramma temporale mostrato nella figura seguente, illustrando l'andamento dei segnali del BUS sincrono di un processore in una situazione dove una periferica P effettua una richiesta d'interruzione (interrupt).

Il bus di controllo contiene i due segnali INTREQ e INTACK per la gestione del meccanismo d'interruzione (le interruzioni generate da P sono di tipo vettorizzato) e il segnale READ_MEM emesso dal processore, che contrassegna un indirizzo valido per un'operazione di lettura da memoria.

Nel diagramma temporale sono indicati solo i cicli di clock e l'istante di tempo in cui la periferica P invia al processore la richiesta d'interruzione; il resto del diagramma va completato.



Esercizio 1 - Seconda Parte

I segnali di controllo INTREQ, INTACK e READ_MEM sono di tipo attivo alto. I segnali INTREQ e INTACK osservano il protocollo di handshake: vengono attivati e disattivati l'uno in risposta all'altro, ma tenendo conto della presenza del clock come si dice più avanti. Si considerino le seguenti specifiche ulteriori:

- Il ciclo di clock è demarcato dal fronte di discesa del clock e il periodo di clock è di 100 ns. Tutti i segnali vengono acquisiti (letti) entro 10 ns (nanosecondi) a partire dal fronte di salita del clock e vengono emessi (scritti) entro 10 ns a partire dal fronte di discesa del clock.
- A partire da quando si accorge che l'interruzione richiesta è stata accettata da parte del processore, la periferica P abbisogna di 50 ns prima di potere emettere sul bus dati il suo vettore d'interruzione, e poi lo mantiene stabile sul bus dati per 100 ns.
- Il vettore d'interruzione rappresenta direttamente l'indirizzo dell'istruzione macchina iniziale della routine di servizio d'interruzione. Nel ciclo successivo alla lettura del vettore, il processore emette sul bus indirizzi l'indirizzo della prima istruzione della routine di servizio d'interruzione, attiva il segnale di controllo READ_MEM e si dispone a leggere da memoria la prima istruzione della routine.
- A partire da quando si accorge che è in corso un'operazione di lettura, la memoria abbisogna di 50 ns prima di potere emettere sul bus dati la parola da leggere e poi la mantiene stabile per 100 ns.
- I segnali sul bus indirizzi vengono portati a riposo e il segnale di controllo READ_MEM viene disattivato non appena i segnali sul bus dati non sono più stabili.

Completando il diagramma temporale è preferibile indicare anche le relazioni di causa-effetto tra le transizioni dei vari segnali, laddove esse esistono, come elemento di maggiore chiarezza.

Esercizio 2 - Prima parte

Si vuole realizzare un circuito combinatorio a quattro ingressi (a, b, c, d) e un'uscita F, caratterizzato dai *mintermini* seguenti (0, 2, 5, 6, 8, 9, 10, 13, 15).

- A. Si scriva la prima forma canonica di F:
- B. Si disegni la mappa di Karnaugh della funzione evidenziando tutti gli implicanti primi
- C. Si sintetizzi la funzione tramite il metodo delle mappe di Karnaugh e si scriva la forma minima come somma di prodotti (SOP). Qualora esistano più forme minime le si indichino tutte (il numero di righe date sotto non è significativo)
- D. Si disegni una rete combinatoria che realizza la funzione calcolata al punto precedente, utilizzando solamente porte a due ingressi.
- E. Si calcoli il ritardo della rete disegnata al punto precedente supponendo che ogni porta AND a due ingressi abbia ritardo di 7 ns e che ogni porta OR a due ingressi abbia ritardo pari a 5 ns (non si consideri il ritardo delle porte NOT applicate agli ingressi primari).

Esercizio 2 - Seconda Parte

Si consideri la funzione booleana di 3 variabili $G(a, b, c)$ espressa dall'equazione seguente:

$$G(a, b, c) = ac + !(a + b + !c) + !abc + ab$$

Si trasformi - tramite le proprietà dell'algebra di commutazione - l'equazione di G in modo da ridurre il costo della sua realizzazione, indicando le singole operazioni svolte e il nome oppure la forma della proprietà utilizzata.

Esercizio 3

Sia dato un circuito sequenziale caratterizzato da:

- un ingresso **I** e un'uscita **U**
- due **bistabili Master/ Slave** di tipo D: (**D1, Q1**) e (**D2, Q2**), con Di ingresso del bistabile e Qi stato / uscita del bistabile

Le funzioni di assegnamento di un nuovo valore ai bistabili e all'uscita U sono:

$$D1 = Q2 \text{ or } (Q1 \text{ and } I)$$

$$D2 = (I \text{ xor } Q1) \text{ or not } Q2$$

$$U = (Q1 \text{ and not } Q2) \text{ or } (I \text{ and } Q2)$$

Si ricorda che l'operatore xor vale 1 se e solo se un solo ingresso è a 1.

Si chiede di disegnare il circuito sopra descritto.

Si chiede di completare il diagramma temporale riportato a pagina seguente. Si noti che:

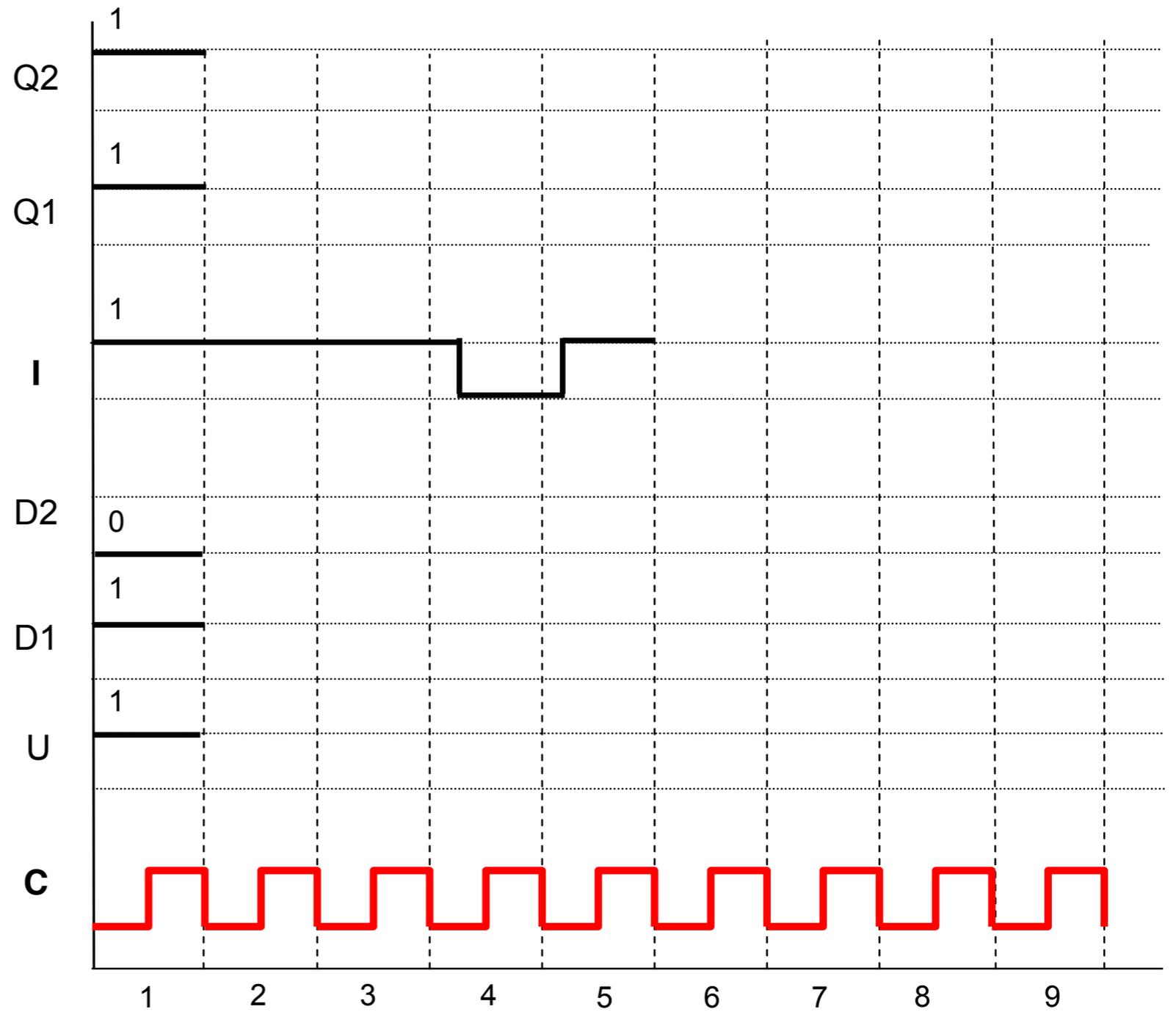
- si devono trascurare completamente il ritardo delle porte e il ritardo di commutazione dei bistabili
- i bistabili sono di tipo "master-slave", in cui l'uscita commuta sul fronte di discesa del clock
- l'ingresso I può variare dopo un fronte di discesa del clock, ma prima del fronte di salita

Esercizio 3

$$D1 = Q2 \text{ or } (Q1 \text{ and } I)$$

$$D2 = (I \text{ xor } Q1) \text{ or not } Q2$$

$$U = (Q1 \text{ and not } Q2) \text{ or } (I \text{ and } Q2)$$



Esercizio 4

Si consideri l'andamento dei segnali di *bus sincrono* quando una periferica *legge* in DMA una parola da memoria. Il DMA è realizzato con arbitraggio centralizzato e l'arbitro di bus è il processore stesso.

Ecco le fasi logiche dell'operazione: la periferica richiede il bus; il processore glielo cede; la periferica lo prende; legge la parola da memoria; e lo restituisce al processore.

Ecco le specifiche del bus:

struttura del bus

- segnali di arbitraggio: busrequest, busgrant e busbusy
- segnali di controllo: read, write e wait
- indirizzo, dato e naturalmente il clock

caratteristiche logiche e temporali del bus

- i segnali di arbitraggio e controllo sono attivi bassi
- il ciclo di clock ha periodo di 200 ns
- i tempi di propagazione sul bus variano tra 20 e 40 ns

protocollo di bus per il trasferimento di parola

- le unità funzionali (master e slave) collegate al bus acquisiscono i segnali sul fronte di salita del clock e li emettono su quello di discesa
- il trasferimento di una parola tra master e slave richiede almeno due cicli: nel primo ciclo vengono attivati segnali di controllo e indirizzo, nel secondo la parola viene trasferita
- se le tempistiche dell'unità slave impongono più di due cicli, l'unità slave proroga il primo ciclo inserendo cicli aggiuntivi tra il primo e il secondo

tempistiche di periferica e memoria

- la periferica prende il bus a seguito dell'attivazione di busgrant e lo restituisce quando il trasferimento termina completamente; in questo intervallo di tempo essa è master
- la memoria si attiva non appena acquisisce i segnali di controllo e riconosce l'indirizzo; dopo di ciò necessita di non meno di 200 ns per essere pronta a trasferire la parola

Si chiede di completare il diagramma temporale mostrato nella figura seguente.

Esercizio 5

Si consideri l'andamento dei segnali di *bus sincrono* quando una periferica *legge* in DMA una parola da memoria. Il DMA è realizzato con arbitraggio centralizzato e l'arbitro di bus è il processore stesso.

Ecco le fasi logiche dell'operazione: la periferica richiede il bus; il processore glielo cede; la periferica lo prende; legge la parola da memoria; e lo restituisce al processore.

Ecco le specifiche del bus:

struttura del bus

- segnali di arbitraggio: busrequest, busgrant e busbusy
- segnali di controllo: read, write e wait
- indirizzo, dato e naturalmente il clock

caratteristiche logiche e temporali del bus

- i segnali di arbitraggio e controllo sono attivi bassi
- il ciclo di clock ha periodo di 200 ns
- i tempi di propagazione sul bus variano tra 20 e 40 ns

protocollo di bus per il trasferimento di parola

- le unità funzionali (master e slave) collegate al bus acquisiscono i segnali sul fronte di salita del clock e li emettono su quello di discesa
- il trasferimento di una parola tra master e slave richiede almeno due cicli: nel primo ciclo vengono attivati segnali di controllo e indirizzo, nel secondo la parola viene trasferita
- se le tempistiche dell'unità slave impongono più di due cicli, l'unità slave proroga il primo ciclo inserendo cicli aggiuntivi tra il primo e il secondo

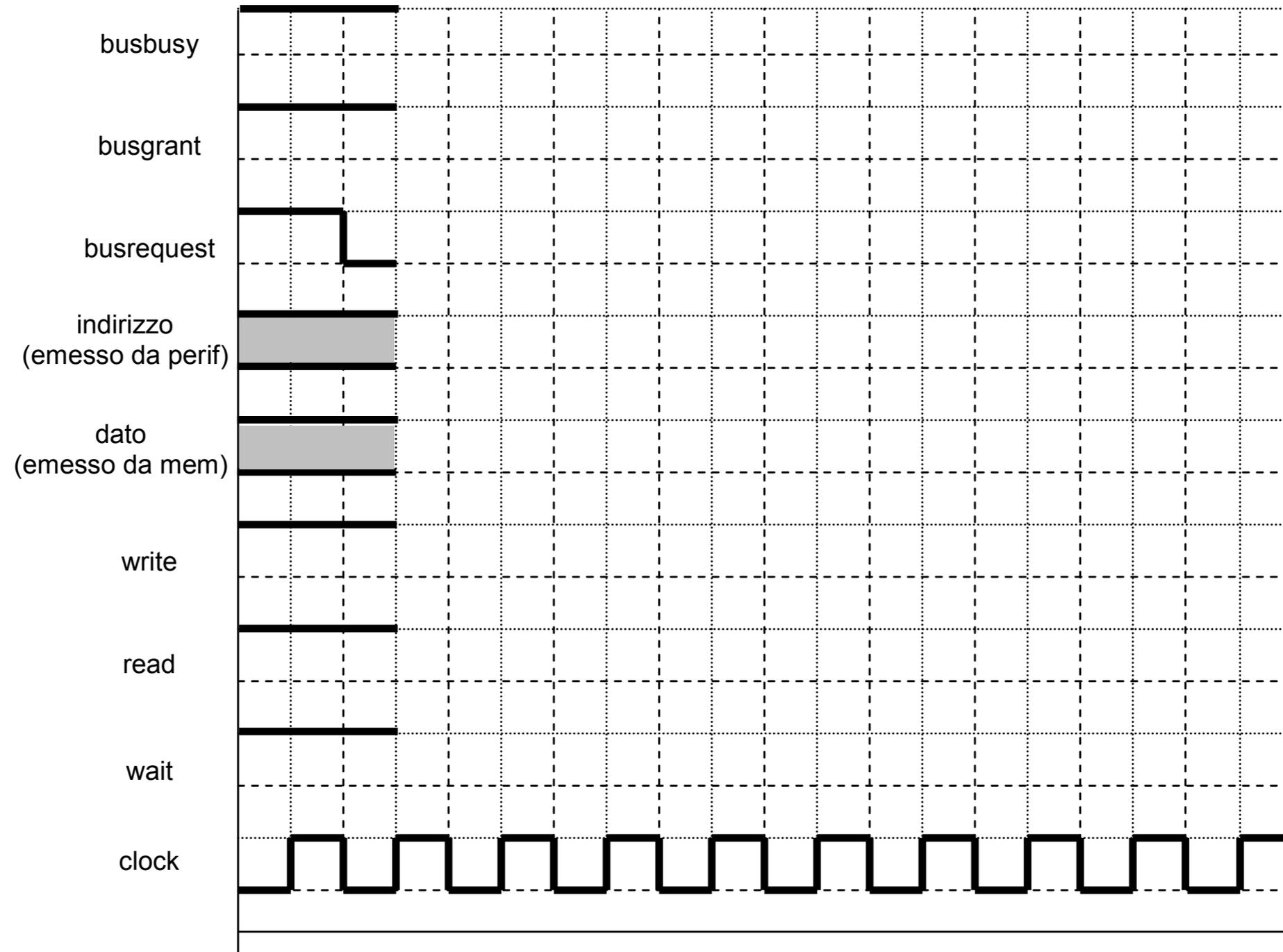
tempistiche di periferica e memoria

- la periferica prende il bus a seguito dell'attivazione di busgrant e lo restituisce quando il trasferimento termina completamente; in questo intervallo di tempo essa è master
- la memoria si attiva non appena acquisisce i segnali di controllo e riconosce l'indirizzo; dopo di ciò necessita di non meno di 200 ns per essere pronta a trasferire la parola

Si chiede di completare il diagramma temporale mostrato nella figura seguente.

Esercizio 5

da completare



Esercizio 6

Sia dato un circuito sequenziale caratterizzato da:

- due ingressi: **I1** e **I2**
- una uscita: **U**
- due **bistabili Master/ Slave** di tipo D: (**D1, Q1**) e (**D2, Q2**), con Di ingresso del bistabile e Qi stato / uscita del bistabile

Le funzioni di assegnamento di un nuovo valore ai bistabili e all'uscita U sono:

$$D1 = \text{not} (I1 \text{ or } I2) \text{ xor } Q2 \quad D2 = Q1 \text{ and } (Q2 \text{ or } I2) \quad U = Q1 \text{ xor not } (Q2)$$

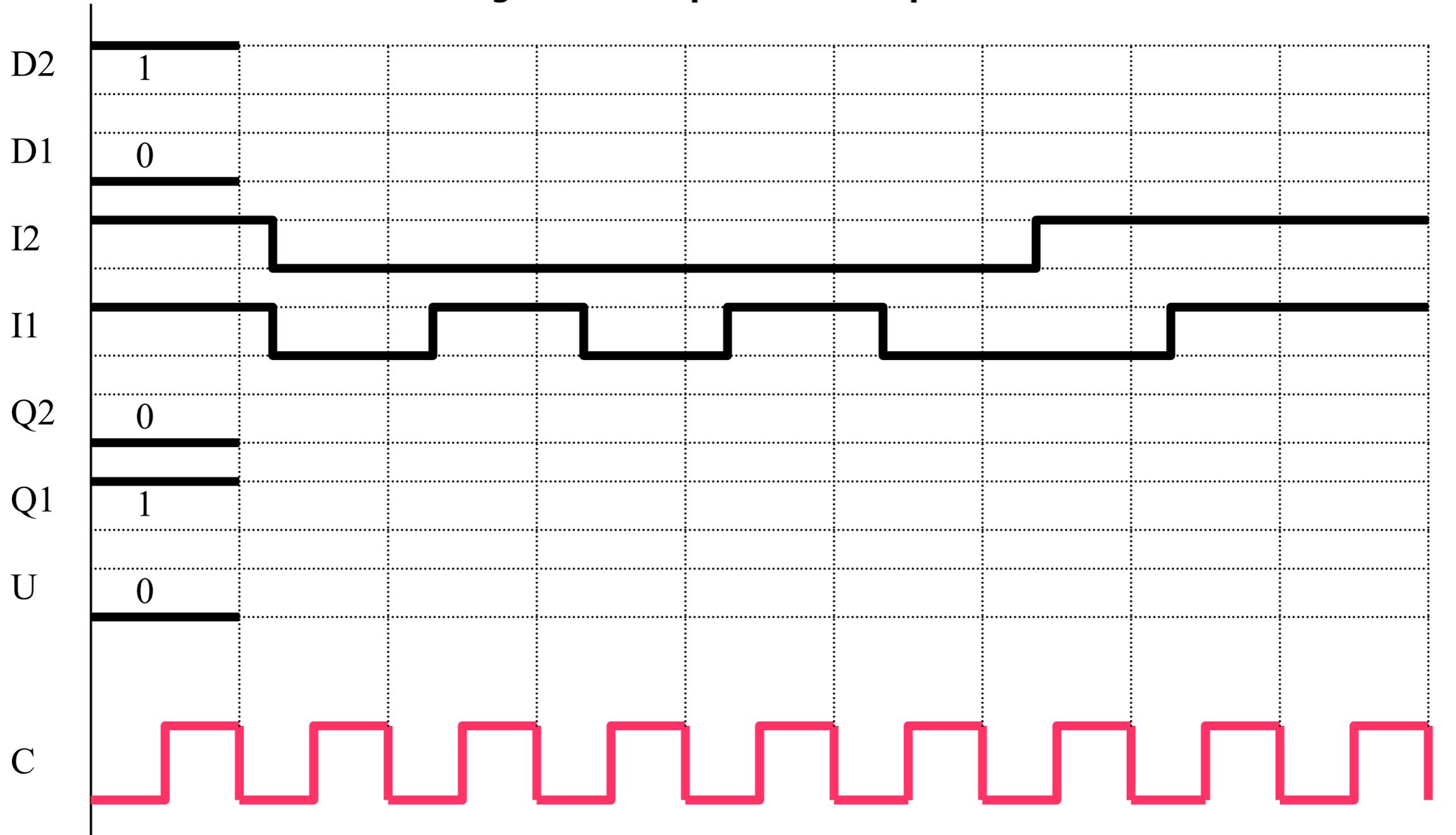
Si chiede di disegnare il circuito sopra descritto.

Si chiede di completare il diagramma temporale riportato a pagina seguente. Si noti che:

- si devono trascurare completamente il ritardo delle porte e il ritardo di commutazione dei bistabili
- i bistabili sono di tipo "master-slave", dove l'uscita commuta sul fronte di discesa del clock
- gli ingressi I1 e I2 possono variare dopo un fronte di discesa del clock, ma prima del fronte di salita

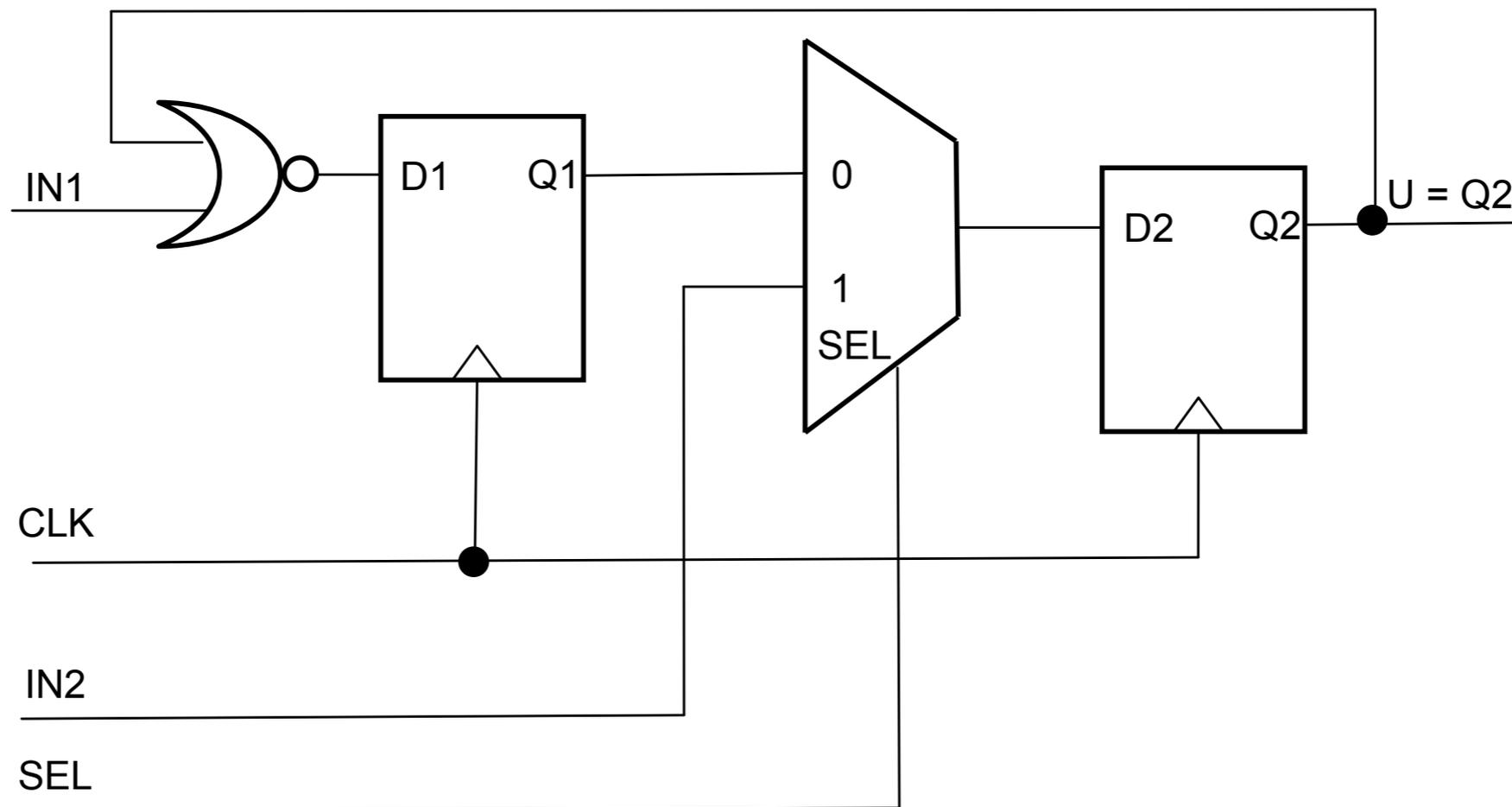
Esercizio 6

diagramma temporale da completare



Esercizio 7

Sia dato il seguente circuito sequenziale:



Esso è composto da due bistabili Master / Slave di tipo D: (D1, Q1) e (D2, Q2), con Di ingresso del bistabile e Qi stato / uscita del bistabile; ed è dotato di tre ingressi IN1, IN2 e SEL, e dell'uscita U.

Si fa notare che il valore del segnale D2 è governato da un selettore (multiplexer), che propaga il segnale Q1 o IN2 in funzione del valore del segnale d'ingresso SEL.

Si chiede di **completare il diagramma temporale** riportato a pagina seguente. Si noti che:

- si devono trascurare completamente i ritardi di propagazione della porta logica NOR e del selettore, e i ritardi di commutazione dei bistabili
- i bistabili sono di tipo "master-slave", la cui uscita commuta sul **fronte di discesa** del clock
- gl'ingressi IN1 e IN2 possono variare in qualunque momento

Esercizio8

seconda parte – logica combinatoria

Si consideri la funzione booleana di quattro variabili $G(a, b, c, d)$ rappresentata dall'espressione seguente:

$$\mathbf{G(a, b, c, d) = !a !(b + c) + c !(a + b) + !a !c !(b + d) + c !b !(a + d)}$$

Si trasformi – tramite le proprietà dell'algebra di commutazione – l'espressione di G in modo da ridurre il costo della sua realizzazione, indicando per nome la singola trasformazione svolta oppure la forma della proprietà utilizzata. Allo scopo si usi la tabella seguente (il numero di righe non è significativo).

Alla prossima lezione

alessandro.nacci@polimi.it