
Calcolatori Elettronici

Reti Sequenziali Asincrone

Ing. dell'Automazione
A.A. 2011/12
Gabriele Cecchetti

Reti Sequenziali Asincrone

- **Sommario:**
 - Circuito sequenziale e bistabile
 - Definizione di rete sequenziale asincrona
 - Condizioni di pilotaggio
 - Modelli strutturali
 - Il Flip-Flop SR asincrono
 - Il riconoscitore di sequenza
- **Riferimenti**
 - G. Corsini “Dalle porte AND OR NOT al sistema calcolatore: un viaggio nel mondo delle reti logiche”: cap. “Reti Sequenziali Asincrone”

definizione di circuito sequenziale
definizione di bistabile

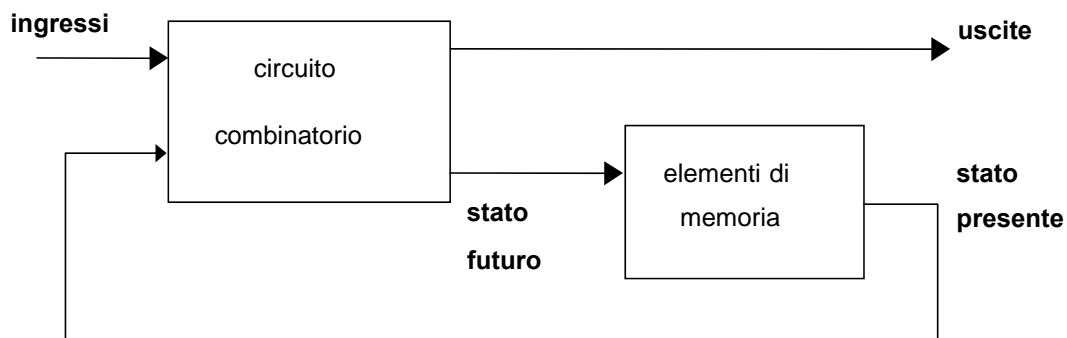
CIRCUITO SEQUENZIALE E BISTABILE

Circuito Sequenziale

- **Il circuito digitale è di tipo sequenziale se le uscite dipendono non solo dai valori correnti degli ingressi, ma anche da (alcuni di) quelli passati:**
 - *una stessa configurazione di ingresso applicata in due istanti di tempo successivi può produrre due valori di uscita differenti.*
- **Il circuito digitale sequenziale (o rete sequenziale) è pertanto dotato, in ogni istante di tempo, di uno stato che, insieme ai valori degli ingressi, ne determina il comportamento futuro:**
 - *lo stato del circuito sequenziale rappresenta una forma di memoria e contiene una sorta di descrizione della storia passata del circuito stesso.*
- **L'elemento funzionale elementare per la realizzazione del circuito sequenziale è il bistabile (elemento di memoria), che è in grado di memorizzare un bit di informazione.**

Struttura

- Il circuito sequenziale è costituito dagli elementi seguenti:
 - *bistabile*, che ha la funzione di memorizzare un bit,
 - *rete combinatoria*, che ha funzione di elaborare le informazioni.
- In ogni istante il circuito sequenziale ha uno stato: il valore dei bit memorizzati nei bistabili facenti parte del circuito.



Elemento di Memoria

- L'elemento di memoria fondamentale, o bistabile, è caratterizzato da due stati (0 e 1) stabili.
 - Mantiene lo stato finché uno o più segnali di ingresso forzano il cambiamento di stato.
- I bistabili sono classificati come segue:
 - secondo il numero di ingressi previsti per comandare il bistabile, e
 - secondo come tali ingressi determinano il cambiamento di stato.

Classificazione dei bistabili

- **bistabile asincrono**: è privo di segnale di sincronizzazione e **cambia stato rispondendo direttamente a eventi** (fronti di transizione) **sui segnali di ingresso**
- **bistabile sincrono**: è **sensibile a un segnale di sincronizzazione** (o di controllo) e **la transizione di stato può avvenire solo in corrispondenza di eventi** (fronti) **sul segnale di controllo**:
 - si può dire che il comportamento di un circuito sincrono viene osservato in istanti discreti di tempo,
 - il segnale di sincronizzazione tipicamente utilizzato è il clock.

Ulteriore classificazione dei bistabili sincroni

- **bistabile trasparente (latch)**
- **flip-flop (bistabile non trasparente)**
 - flip flop master-slave (a livello)
 - flip flop edge-triggered (a fronte)

Definizione

Modello strutturale

INTRODUZIONE ALLE RETI SEQUENZIALI ASINCRONE

Definizione di Rete Sequenziale Asincrona

- Una rete sequenziale asincrona è una qualunque struttura che soddisfa ai seguenti requisiti:
 - è dotata di N var. d'ingresso $x_{N-1}, x_{N-2}, \dots, x_0$
 - è dotata di M var. di uscita $z_{M-1}, z_{M-2}, \dots, z_0$
 - è dotata di un **meccanismo di marcatura** che seleziona ad ogni istante uno ed un solo elemento appartenente ad un opportuno insieme $S = \{S^0, S^1, \dots, S^{K-1}\}$
 - implementa una **legge per gli stati interni** $A: S \times X \Rightarrow S$
 - implementa una **legge per gli stati di uscita** $B: S \Rightarrow Z$
 - ottempera alla seguente **legge di evoluzione del tempo**:
“se X e S sono lo stato d'ingresso presente e lo stato interno marcato ad un certo istante, far coincidere lo stato di uscita con $B(S)$; individuare inoltre lo stato interno successivo $A\{S, X\}$ e marcarlo e così via all'infinito.

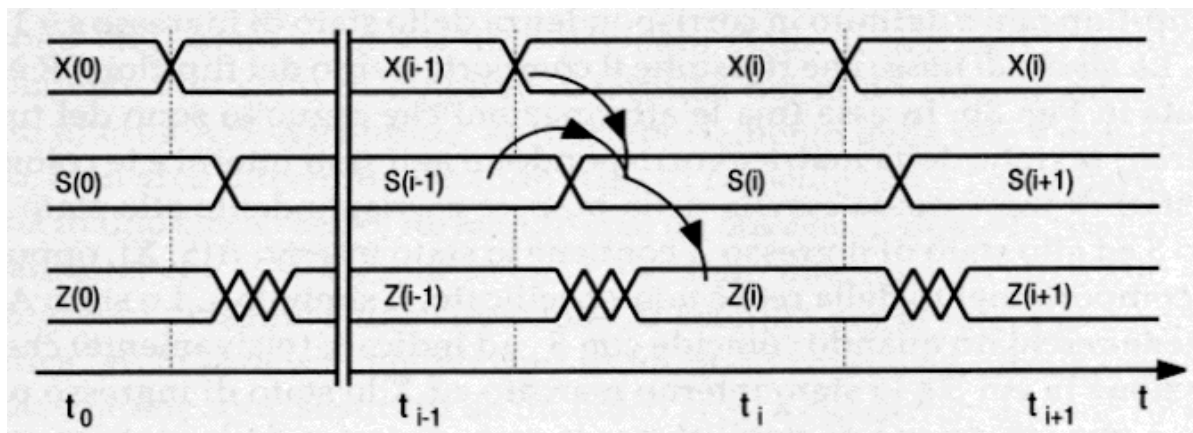
Considerazioni sulle RSA

- Le RSA sono sempre in evoluzione.
- Quando uno stato interno successivo coincide con quello attualmente marcato siamo in una situazione di stabilità.
- Le RSA significative sono quelle in cui la legge A è tale da assicurare che, partendo da una situazione di stabilità, un'altra situazione di stabilità è sempre raggiunta, quale che sia lo stato di ingresso che viene ricevuto dalla rete.
 - **Condizione sufficiente:**
 $A\{S,X\}$ coincide con S o con $A\{A\{S,X\},X\}$ (dette *leggi normali*).

Condizioni di pilotaggio per le RSA

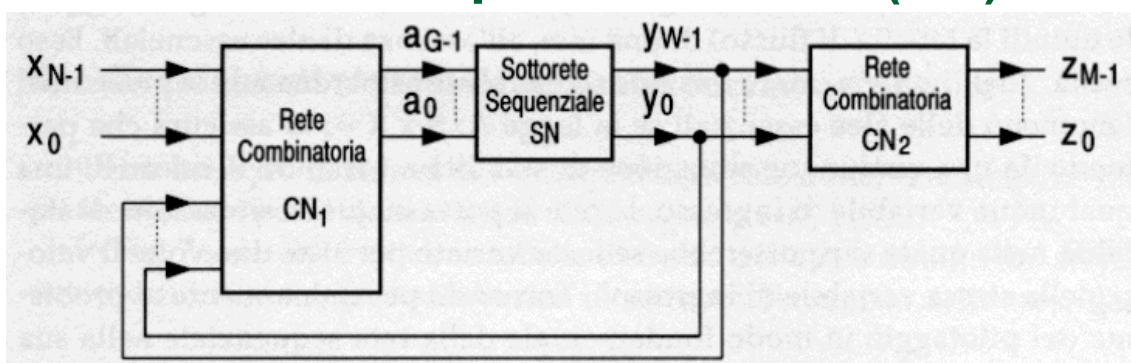
- Data una situazione di stabilità iniziale, **cambiare lo stato d'ingresso solo quando essa è in situazione di stabilità** (*pilotaggio in modo fondamentale*).
- ↑ *Evita il rincorrersi di transitori che renderebbero indeterminato lo stato interno successivo.*
- Far sì che ogni variazione dello stato di ingresso derivi dalla **transizione di una sola var. di ingresso** (*pilotaggio senza transizioni multiple in ingresso*).
- ↑ *Impedisce che la rete si evolva in modo differente da quello progettato.*

Evoluzione temporale di una R.S.A.



- $Z(i)$ è funzione di $S(i)$,
- $S(i)$ è funzione di $X(i)S(i-1)$, quindi
- $Z(i)$ è funzione di $X(i), X(i-1), \dots, S(0)$.

Modelli strutturali per le R.S.A. (1/2)



- $S = \{S^0, S^1, \dots, S^{K-1}\}$ sono codificati tramite K delle 2^W combinazioni dei valori delle var. di stato $y_{W-1}, y_{W-2}, \dots, y_0$ con $W \geq \lceil \log_2 K \rceil$;
- La sottorete sequenziale SN marca all'uscita lo stato interno;
- $CN_1 + SN$ implementano sia la legge A che il meccanismo di marcatura;
- CN_2 implementa la legge B.

Modelli strutturali per le R.S.A. (2/2)

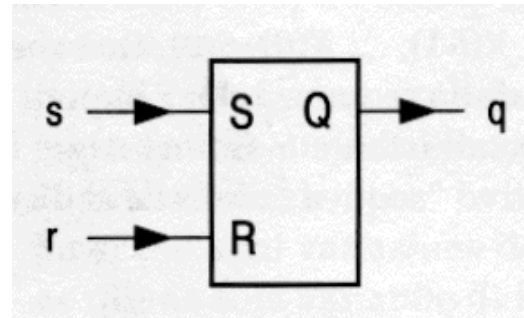
- SN può essere implementata con
 - **ritardi** (corto circuiti), oppure
 - **Flip-Flop SR** (è una *sottorete asincrona elementare*).
- Il meccanismo di marcatura implementato dalla rete SN non è necessariamente primitivo.
 - *Se varia più di un bit in uscita da SN può nascere una corsa delle variabili di stato.*
- Pertanto occorre:
 - ⇒ **codificare CN_1 in modo tale che non abbia alee per evitare che SN marchi stati spuri;**
 - ⇒ **pilotare in modo fondamentale CN_1 (SN marca il nuovo stato interno solo quando CN_1 è a regime) e senza transizioni multiple in ingresso.**

bistabile SR asincrono
definizione struttura e funzionamento

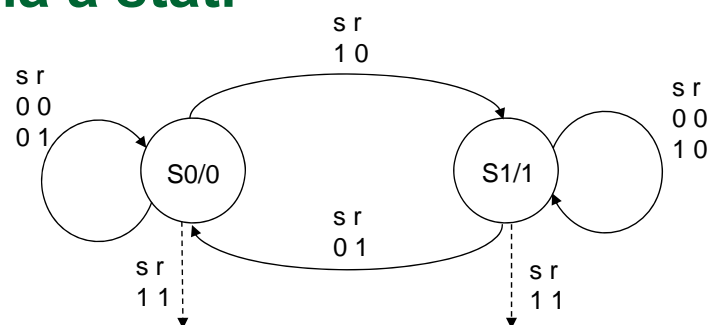
BISTABILE SR ASINCRONO

Flip-Flop SR (elementare): specifica di funzionamento

s	r	q	
1	0	1	Set
0	1	0	Reset
0	0	$q'=q$	<u>Conservazione</u>
1	1	-	Non consentito



Flip-Flop SR (elementare): diagramma a stati



- Gli stati con le orecchiette sono stabili
- Lo stato di ingresso $\langle sr \rangle = 11$ produce un comportamento indeterminato

Flip-Flop SR (elementare): tabella degli stati

sr		00	01	11	10	q
		S0	S0	—	S1	
s	S1	S1	S0	—	S1	1

- E' in condizione normale perché per ogni ingresso esiste uno stato stabile

Flip-Flop SR (elementare): sintesi

1. Modello strutturale scelto

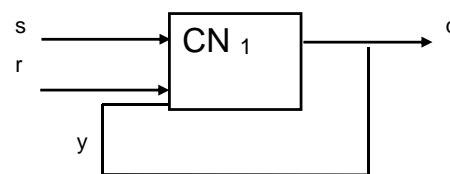
- La sottorete SN è un corto circuito

2. Codifica degli stati

$$S_0 = \langle 0 \rangle$$

$$S_1 = \langle 1 \rangle$$

- Con questa codifica degli stati CN₂ è un corto circuito



3. Tabella degli stati

y sr		00	01	11	10	q
		0	0	-	1	
s	1	1	0	-	1	1

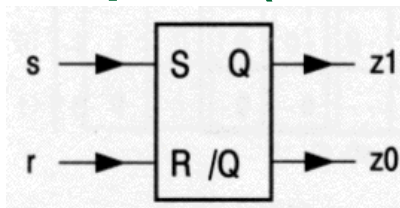
4. Sintesi CN1

$$q = a = s + y/r = /(s/(y/r))$$

Flip-Flop SR (elementare): considerazioni

- Con questa sintesi per l'ingresso $\langle 11 \rangle$ lo stato indeterminato diventa ora S_1 .
- **Se l'ingresso cambia da 11 a 00**, non potendo avvenire la transizione contemporaneamente (la variazione sugli ingressi è 11,10,00 oppure 11,01,00 a seconda di quale delle due variabili è più rapida), **lo stato finale sarà S_1 oppure S_0** .

Flip-Flop SR (reale)

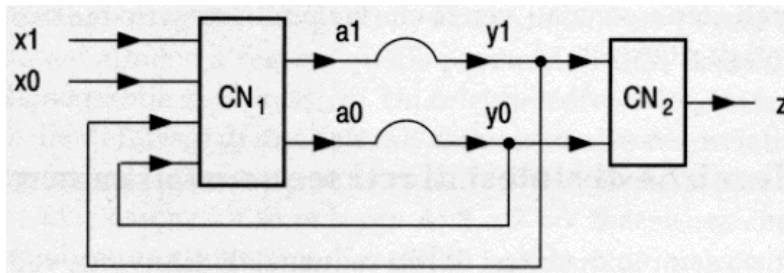


- Il bistabile SR è dotato di due ingressi S (Set) e R (Reset), e di due uscite Q e /Q (forma negata di Q):
 - se $Q = 1$ ($/Q = 0$): stato di set
 - se $Q = 0$ ($/Q = 1$): stato di reset

s r	00	01	11	10	z1	z0
S0	(S0)	(S0)	—	S1	0	1
S1	(S1)	S0	—	(S1)	1	0

- *In questo caso il flip-flop SR ha una seconda uscita il cui valore è il complemento dell'altra uscita.*
- **L'uscita Q rappresenta quindi lo stato memorizzato:**
 - se $S = R = 0$, le uscite Q e /Q possono valere 1 e 0, rispettivamente, ma ...
 - se $S = R = 0$, le uscite Q e /Q possono anche valere 0 e 1, rispettivamente
- *Dunque, a parità di ingressi (cioè $S = R = 0$) l'uscita Q ammette due valori possibili.*

Flip-Flop SR: modello strutturale



Stato interno	Codifica	
	y1	y0
s0	0	1
s1	1	0

- Questa codifica degli stati interni porta ad una corsa delle variabili di stato passando dallo stato S_0 allo stato S_1 , e viceversa. Tale corsa potrebbe portare a marcare uno stato interno codificato come 11 (S_A) o 00 (S_B).

⇒ **Occorre allora modificare la tabella di flusso.**

Flip-Flop SR: tabelle di flusso modificata

- Lo stato ponte S_A consente di passare in modo stabile dallo stato interno S_0 a quello S_1 , e viceversa.
- L'unico effetto indesiderato che rimane è l'uscita 11 (questo però è plausibile visto che non è mai possibile far commutare due variabili contemporaneamente).

s r					z1 z0	
	00	01	11	10		
S0	(S0)	(S0)	—	SA	0	1
S1	(S1)	SA	—	(S1)	1	0
SA	—	S0	—	S1	1	1
SB	—	—	—	—	0	0

Flip-Flop SR: codifica degli stati interni

s r	00	01	11	10	z1	z0
S0	(S0)	(S0)	—	SA	0	1
S1	(S1)	SA	—	(S1)	1	0
SA	—	S0	—	S1	1	1
SB	—	—	—	—	0	0

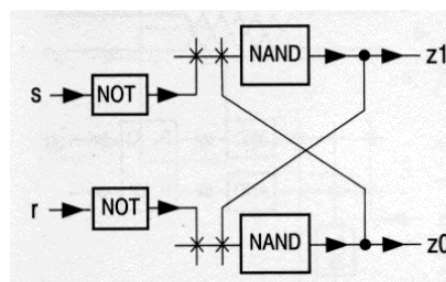
Stato interno	Codifica y1 y0	
S0	0	1
S1	1	0
SA	1	1
SB	0	0

s r	00	01	11	10	z1	z0
01	(01)	(01)	--	11	0	1
10	(10)	11	--	(10)	1	0
11	--	01	--	10	1	1
00	--	--	--	--	0	0

- Da cui si ottiene:

$$z_1 = /y_0 + s = /(y_0/s)$$

$$z_2 = /y_1 + r = /(y_1/r)$$
- Qui a destra viene mostrata la realizzazione tramite porte NAND.

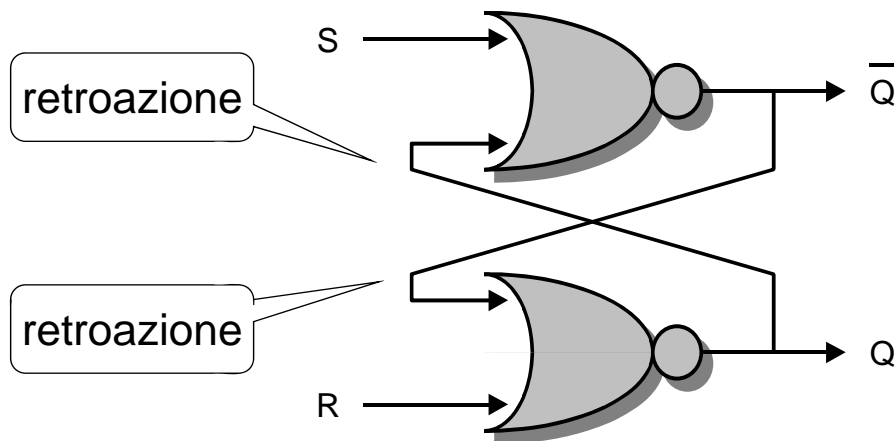


Flip-Flop SR: tabella di flusso finale

- Ora il flip-flop SR accetta, senza che si verifichino malfunzionamenti qualunque stato di ingresso e qualunque transizione multipla delle variabili di ingresso esclusa quella dallo stato di ingresso 11 allo stato di ingresso 00.

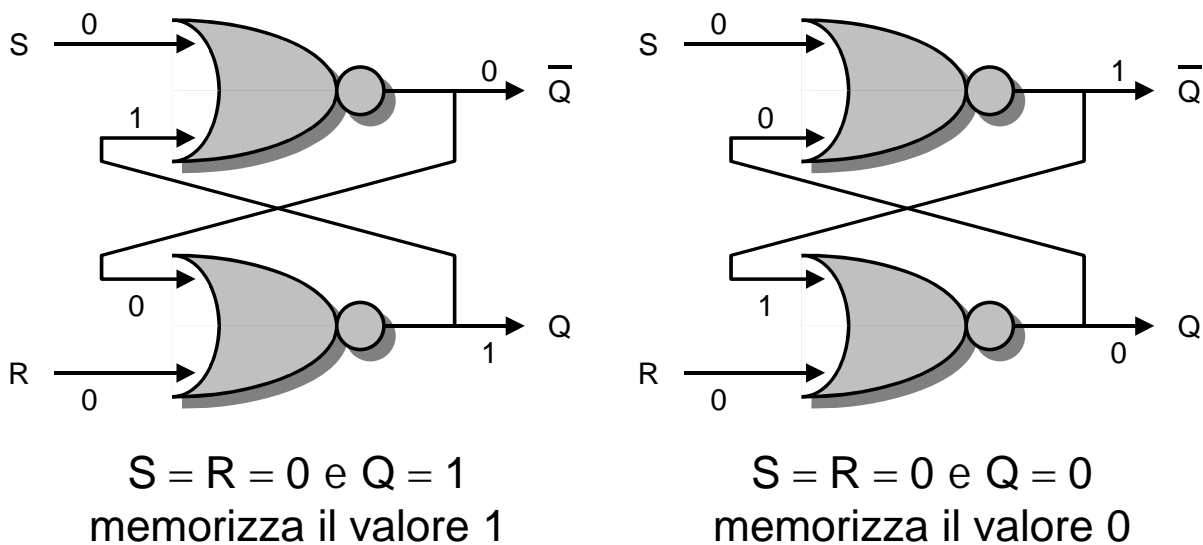
s r	00	01	11	10	z1	z0
S0	(S0)	(S0)	SA	SA	0	1
S1	(S1)	SA	SA	(S1)	1	0
SA	SB	S0	(SA)	S1	1	1
SB	SA	SA	SA	SA	0	0

Come memorizzare il Bit



- **Realizzazione circuitale del Flip-Flop S-R asincrono mediante porte NOR:**
 - il circuito ha due ingressi, S e R, e due uscite, Q e \bar{Q} .

Come funziona il Bistabile SR asincrono (1/2)



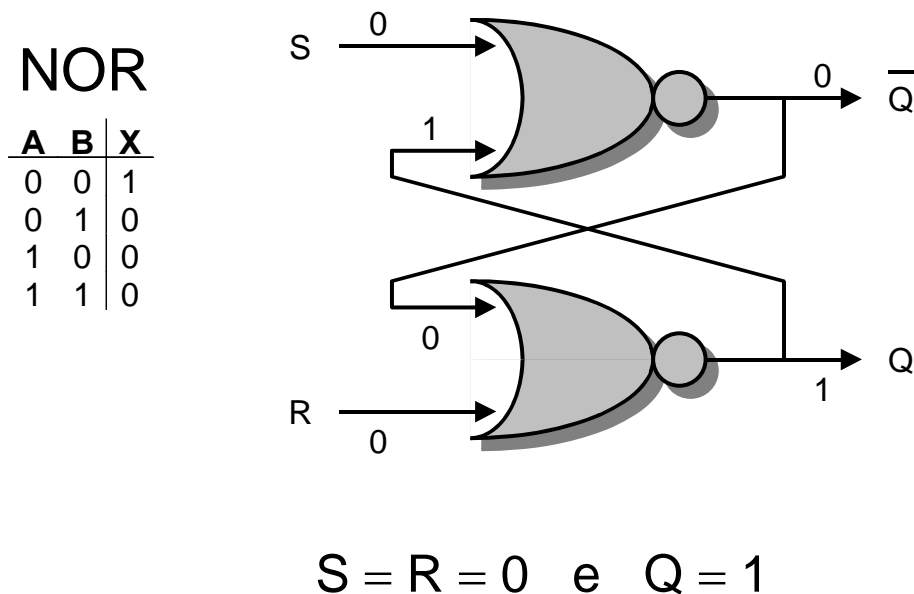
il circuito ha due stati di equilibrio (bistabile)

Come funziona il Bistabile SR asincrono (2/2)

- Il bistabile SR è in grado di memorizzare due valori logici distinti:
 - se $Q = 1$ il bistabile memorizza 1
 - se $Q = 0$ il bistabile memorizza 0
- Stato a 0:
 - se $S = 0$ e $R = 1$, qualunque sia il valore dello stato presente, le uscite Q e \bar{Q} vengono portate a 0 e 1, rispettivamente
 - il nuovo stato è 0
- Stato a 1:
 - se $S = 1$ e $R = 0$, qualunque sia il valore dello stato presente, le uscite Q e \bar{Q} vengono portate a 1 e 0, rispettivamente
 - il nuovo stato è 1

Bistabile SR asincrono: transizione da 1 a 0

(1/4)

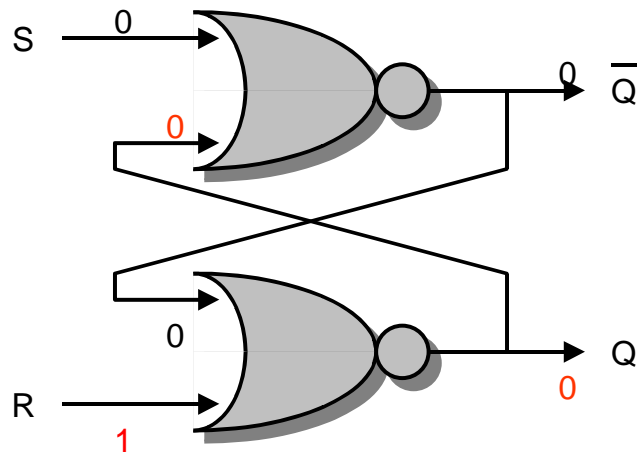


Bistabile SR asincrono: transizione da 1 a 0

(2/4)

NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

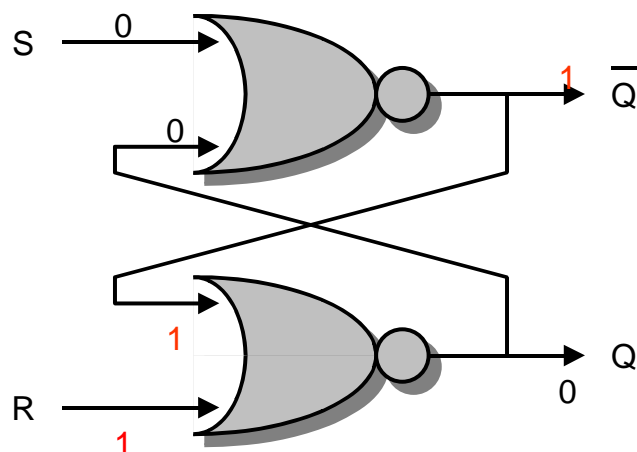


Bistabile SR asincrono: transizione da 1 a 0

(3/4)

NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0



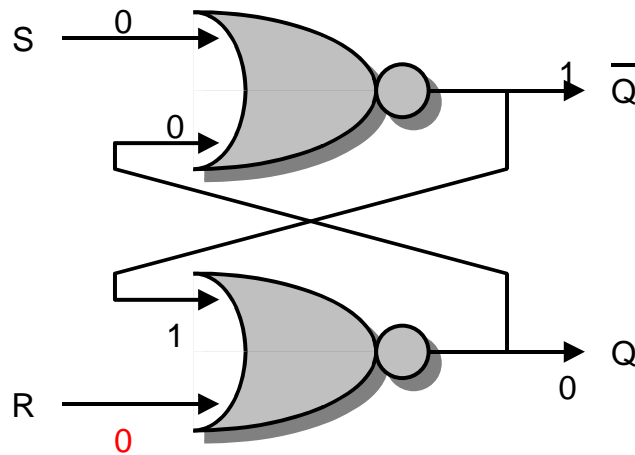
$S = 0$ e $R = 1$ allora Q diventa 0

Bistabile SR asincrono: transizione da 1 a 0

(4/4)

NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

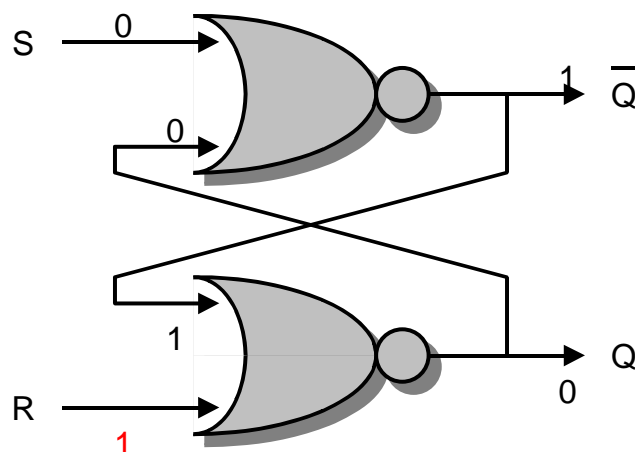


$S = R = 0$ allora Q rimane 0

Bistabile SR asincrono: ingresso di Reset con $Q = 0$

NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

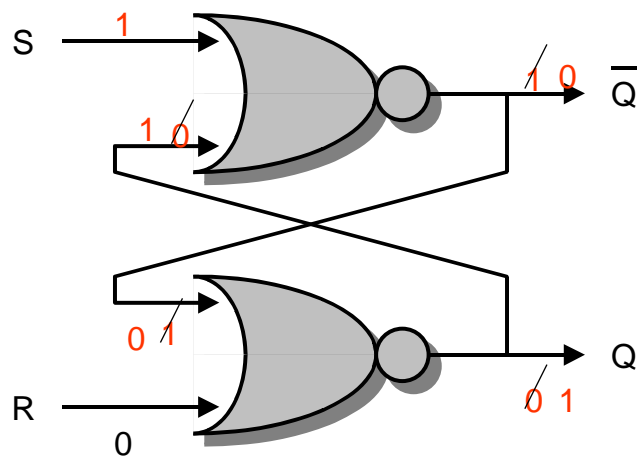


$S = 0$ e $R = 1$ allora Q rimane 0

Bistabile SR asincrono: transizione da 0 a 1

NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

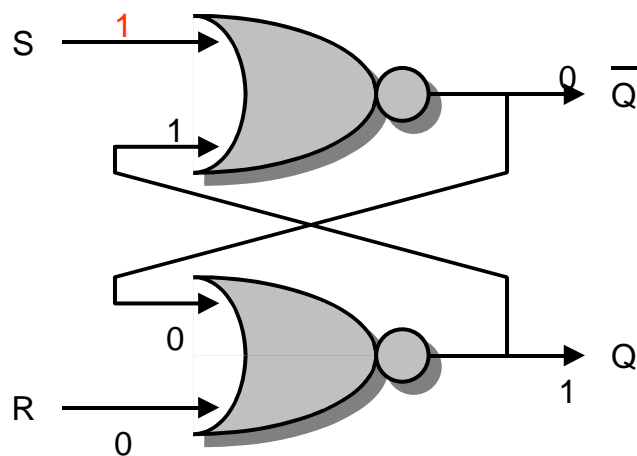


$S = 1$ e $R = 0$ allora Q diventa 1

Bistabile SR asincrono: ingresso di Set con $Q = 1$

NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

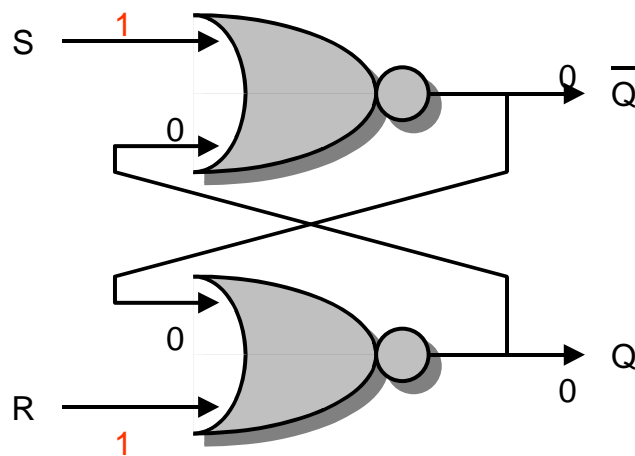


$S = 1$ e $R = 0$ allora Q rimane 1

Bistabile SR asincrono: anomalia per $S = R = 1$

NOR

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0



$S = 1$ e $R = 1$ allora idealmente Q e \bar{Q} a 0

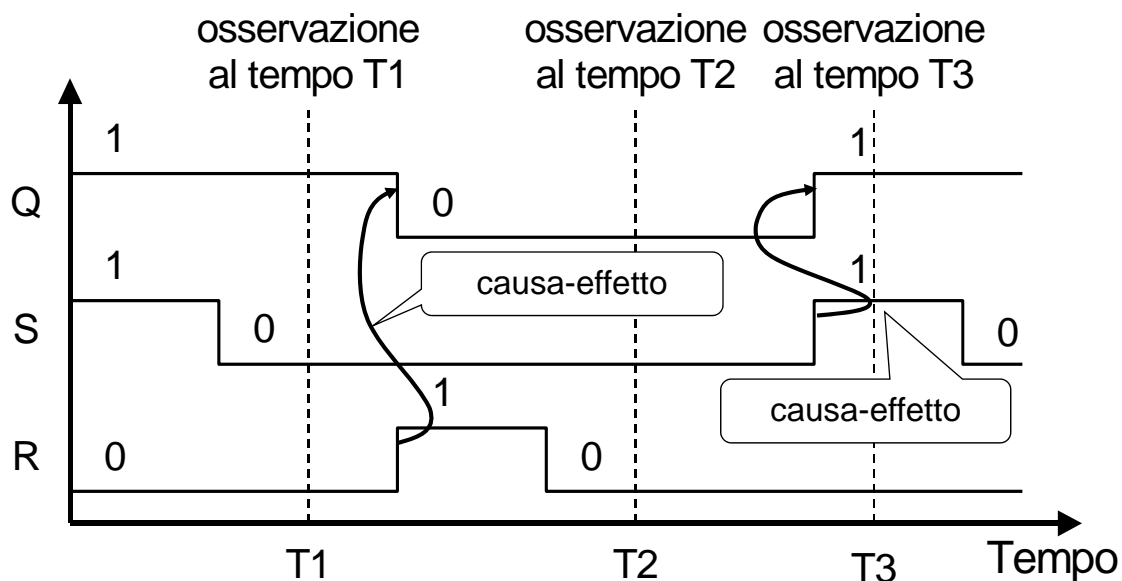
Bistabile SR asincrono: riassunto

- Se $S = R = 0$, l'uscita Q mantiene il valore logico corrente di un bit (cioè 0 oppure 1).
- Se $S = 1$ e $R = 0$, l'uscita Q assume valore 1.
- Se $S = 0$ e $R = 1$, l'uscita Q assume valore 0.
- La configurazione di ingresso $S = R = 1$ è vietata.
- In quest'ultima circostanza il comportamento del bistabile SR non è ben definito.

Diagramma temporale

- Un modo per visualizzare comportamenti di circuiti sequenziali, che dipendono dal tempo e da eventi passati, è il diagramma temporale.
- Il diagramma temporale è un sistema di assi cartesiani con:
 - in ascissa il tempo (in istanti discreti)
 - in ordinata i vari segnali i cui valori logici si succedono al trascorrere del tempo

Diagramma temporale del FF SR asincrono

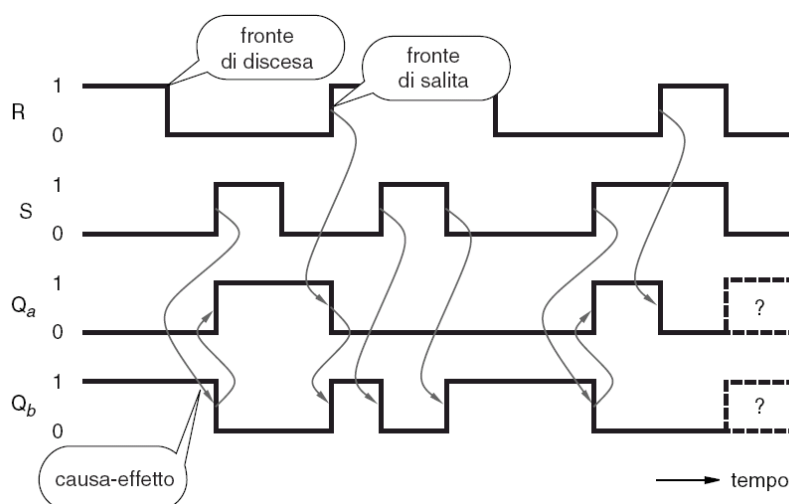


le frecce indicano un rapporto tra i fronti di tipo causa-effetto

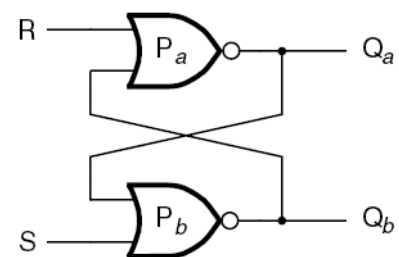
Tabella di Transizione

- Il comportamento del bistabile, e in generale del circuito sequenziale, non è rappresentabile tramite tabella di verità (come la rete combinatoria).
- L'uscita del circuito sequenziale non dipende solo dagli ingressi presenti, ma anche da quelli passati, ovvero anche dallo stato presente del circuito.
- Si può rappresentare il comportamento del circuito sequenziale mediante la tabella di transizione:
 - righe indicizzate tramite le configurazioni di ingresso
 - colonna di uscita contenente valori costanti e anche l'indicazione simbolica dello stato presente

Tabella di Transizione del FF SR asincrono



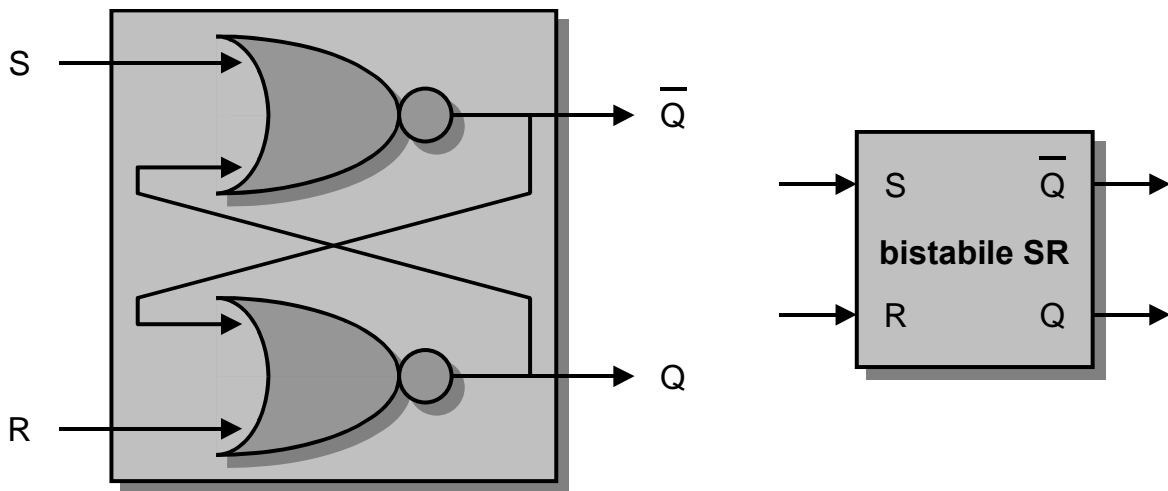
bistabile SR asincrono



S	R	Q_a	Q_b	o anche
0	0	0/1	1/0	$Q_a Q_b$
0	1	0	1	
1	0	1	0	resto idem
1	1	0	0	

tabella di transizione

Rappresentazione del FF SR asincrono

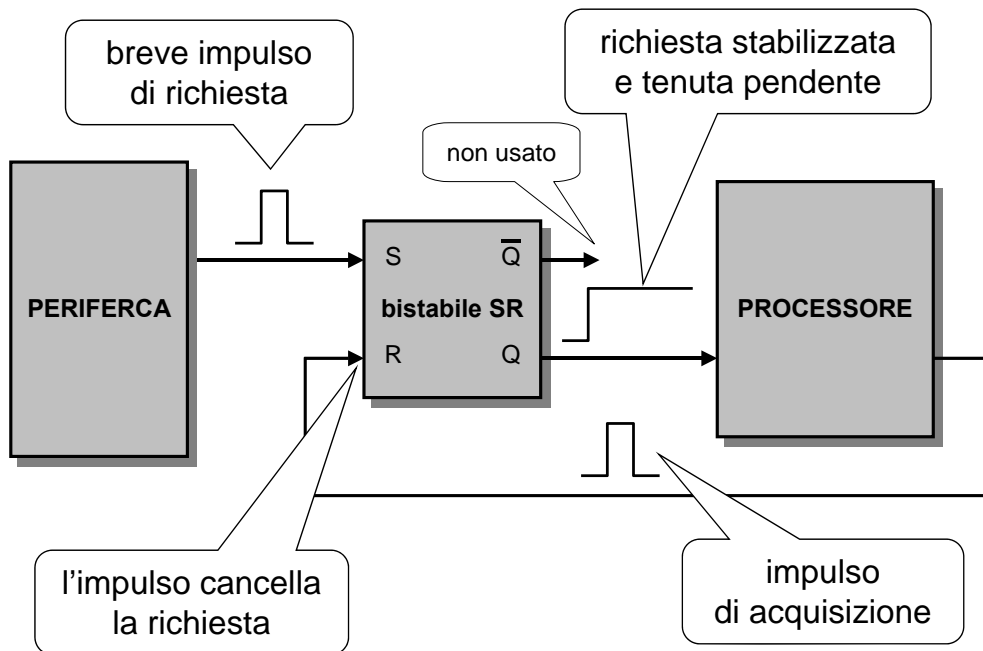


il bistabile SR (set-reset) come blocco funzionale sequenziale

Esempio come Adattatore

- Si supponga di avere una periferica che deve mandare un segnale di richiesta (per esempio una richiesta di interruzione, interrupt) a un processore.
- La periferica genera solo un breve impulso di richiesta, ma il processore potrebbe essere occupato e non in grado di rispondere subito alla richiesta, onorandola.
- È dunque necessario interporre tra periferica e processore un circuito digitale adattatore (interfaccia), che:
 - riceva l'impulso di richiesta proveniente dalla periferica, lo memorizzi, stabilizzandolo, e lo mandi al processore
 - mantenga pendente la richiesta fintantoché il processore non sia disponibile a onorarla
 - cancelli la richiesta, non appena il processore abbia segnalato di averla acquisita e di essere pronto a onorarla

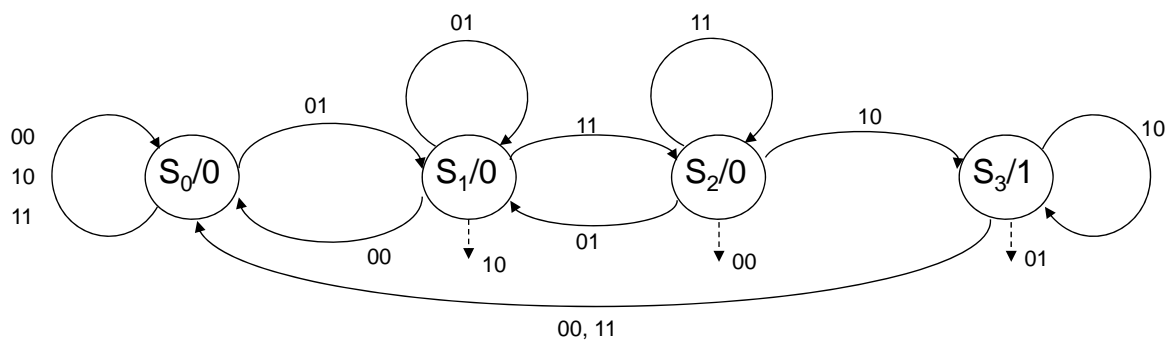
Esempio come Adattatore: schema logico



Riconoscitore di sequenza 01,11,10 (1/3)



■ Diagramma di flusso



Riconoscitore di sequenza 01,11,10 (2/3)

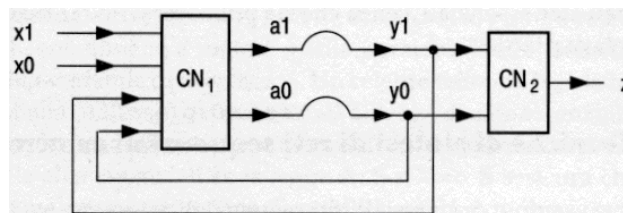
	x1 x0				z
	00	01	11	10	
S0	S0	S1	S0	S0	0
S1	S0	S1	S2	-	0
S2	-	S1	S2	S3	0
S3	S0	-	S0	S3	1

stato interno	codifica y1 y0	
S0	0	0
S1	0	1
S2	1	1
S3	1	0

	x1 x0				z
	00	01	11	10	
00	00	01	00	00	0
01	00	01	11	-	0
11	-	01	11	10	0
10	00	-	00	10	1

1. Tabella di flusso
2. Codifica degli stati interni
3. Tabella di flusso con gli stati interni codificati

Riconoscitore di sequenza 01,11,10 (3/3)



- Modello strutturale
- Leggi caratterizzanti CN1 e CN2

y1 y0	x1 x0			
	00	01	11	10
00	0	0	0	0
01	0	0	1	-
11	-	0	1	1
10	0	-	0	1

$$a_1 = x_1 y_0 + x_1 / x_0 y_1$$

y1 y0	x1 x0			
	00	01	11	10
00	0	1	0	0
01	0	1	1	-
11	-	1	1	0
10	0	-	0	0

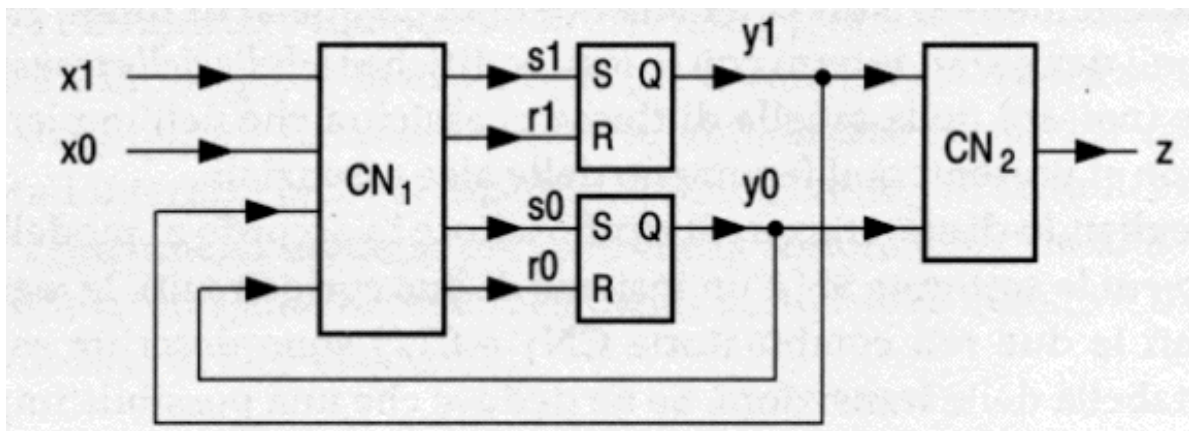
$$a_0 = /x_1 x_0 + x_0 y_0$$

y1 y0	z
0 0	0
0 1	0
1 1	0
1 0	1

$$z = y_1 / y_0$$

Riconoscitore di sequenza 01,11,10 implementato con flip-flop SR (1/2)

Modello strutturale



Riconoscitore di sequenza 01,11,10 implementato con flip-flop SR (2/2)

1. Tabella degli stati codificati
2. Tabella di applicazione del flip-flop SR
3. Tabelle delle variabili di pilotaggio dei 2 flip-flop
4. Tabella delle rete CN₂

1

x1 x0	00	01	11	10	z
00	00	01	00	00	0
01	00	01	11	-	0
11	-	01	11	10	0
10	00	-	00	10	1

2

valore attuale della variabile	nuovo valore della variabile	nuovi valori delle variabili	
y _w	y _w	s _w	r _w
0	0	0	-
0	1	1	0
1	1	-	0
1	0	0	1

3

x1 x0	00	01	11	10
00	0-	0-	0-	0-
01	0-	0-	10	--
11	--	01	-0	-0
10	01	--	01	-0

s1 r1

$$s1 = x1*y0; \quad r1 = /x1 + x0*/y0$$

3

x1 x0	00	01	11	10
00	0-	10	0-	0-
01	01	-0	-0	--
11	--	-0	-0	01
10	0-	--	0-	0-

s0 r0

$$s0 = /x1*x0; \quad r0 = /x0$$

4

y1 y0	z
0 0	0
0 1	0
1 1	0
1 0	1

$$z = y1*/y0$$