



# Architetture degli Elaboratori e delle Reti I

Laboratorio - linea 3 (M-Z)

8

## Macchine a stati finiti (2)

# Esempio MSF : semaforo

## **Problema:**

Il dipartimento per la gestione del traffico stradale necessita di un nuovo sistema per la gestione dei semafori (modello europeo standard). Dobbiamo disegnare un circuito sincrono basato su una macchina di Moore che gestisce un nuovo tipo di semafori in due diversi tipi di incrocio stradale (traffico normale, traffico intenso). Dovranno essere gestite sei luci : Rosso, Giallo e Verde secondo la direzione Nord→Sud e Rosso, Giallo e Verde secondo la direzione Est→Ovest.

# Esempio MSF : semaforo

## Problema:

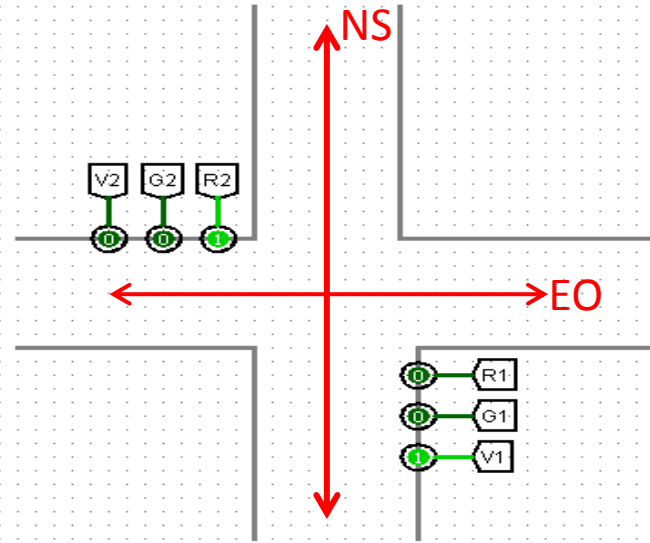
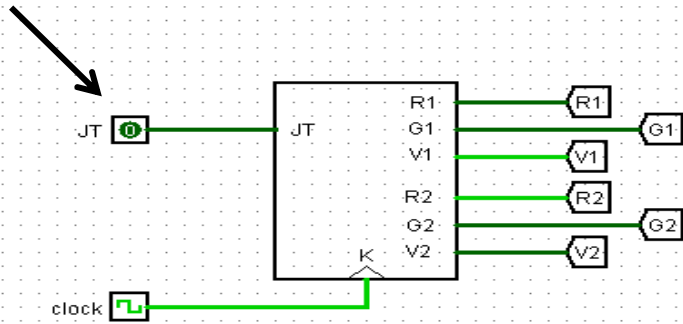
Il dipartimento per la gestione del traffico stradale necessita di un nuovo sistema per la gestione dei semafori (modello europeo standard). Dobbiamo disegnare un circuito sincrono basato su una macchina di Moore che gestisce un nuovo tipo di semafori in due diversi tipi di incrocio stradale (traffico normale, traffico intenso). Dovranno essere gestite sei luci : Rosso, Giallo e Verde secondo la direzione Nord→Sud e Rosso, Giallo e Verde secondo la direzione Est→Ovest.

## scheduling

Traffico normale:	Traffico intenso:
<b>direzione</b> ↖ (NS) → (EO) ↗	(NS) (EO)
R V	R V
R G	R G
V R	R R
G R	V R
	G R
	R R

# Esempio MSF : semaforo

Junction type (JT): 1 → traffico moderato



**Traffico normale:** (JT=1)

**Traffico intenso:** (JT=0)

**direzione**

(NS) → (EO)

R<sub>1</sub>      V<sub>2</sub>

R<sub>1</sub>      G<sub>2</sub>

V<sub>1</sub>      R<sub>2</sub>

G<sub>1</sub>      R<sub>2</sub>

(NS)      (EO)

R<sub>1</sub>      V<sub>2</sub>

R<sub>1</sub>      G<sub>2</sub>

R<sub>1</sub>      R<sub>2</sub>

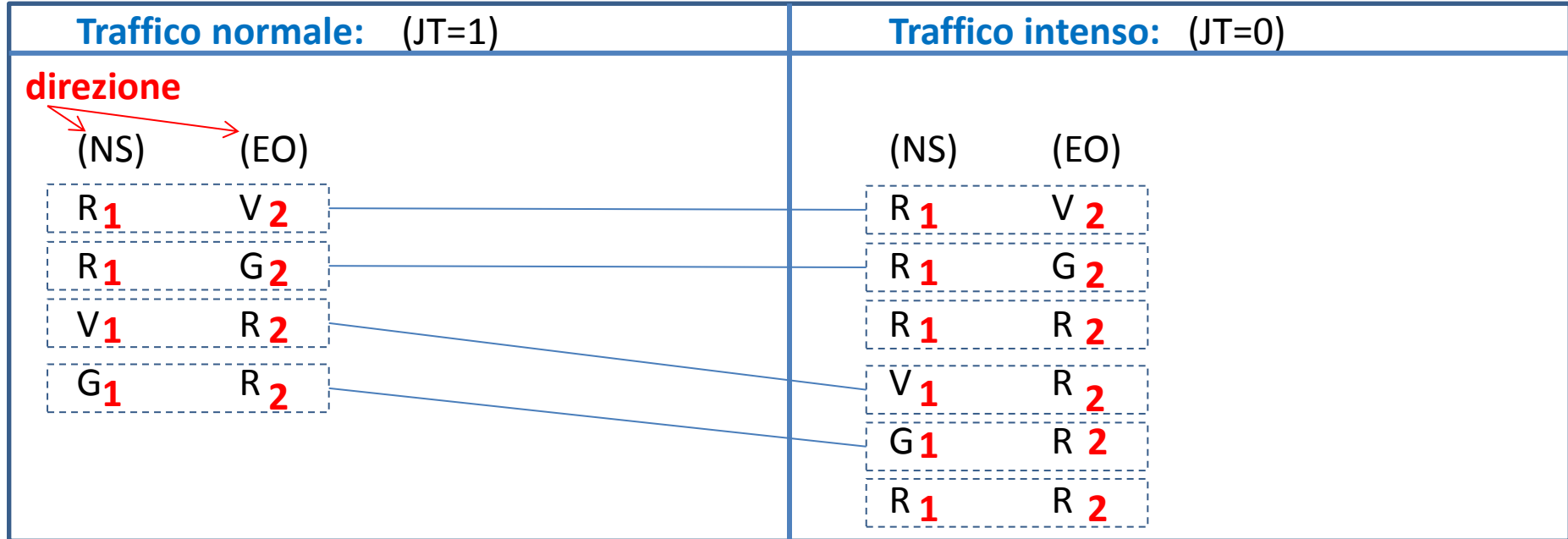
V<sub>1</sub>      R<sub>2</sub>

G<sub>1</sub>      R<sub>2</sub>

R<sub>1</sub>      R<sub>2</sub>

# Esempio MSF : semaforo

scheduling

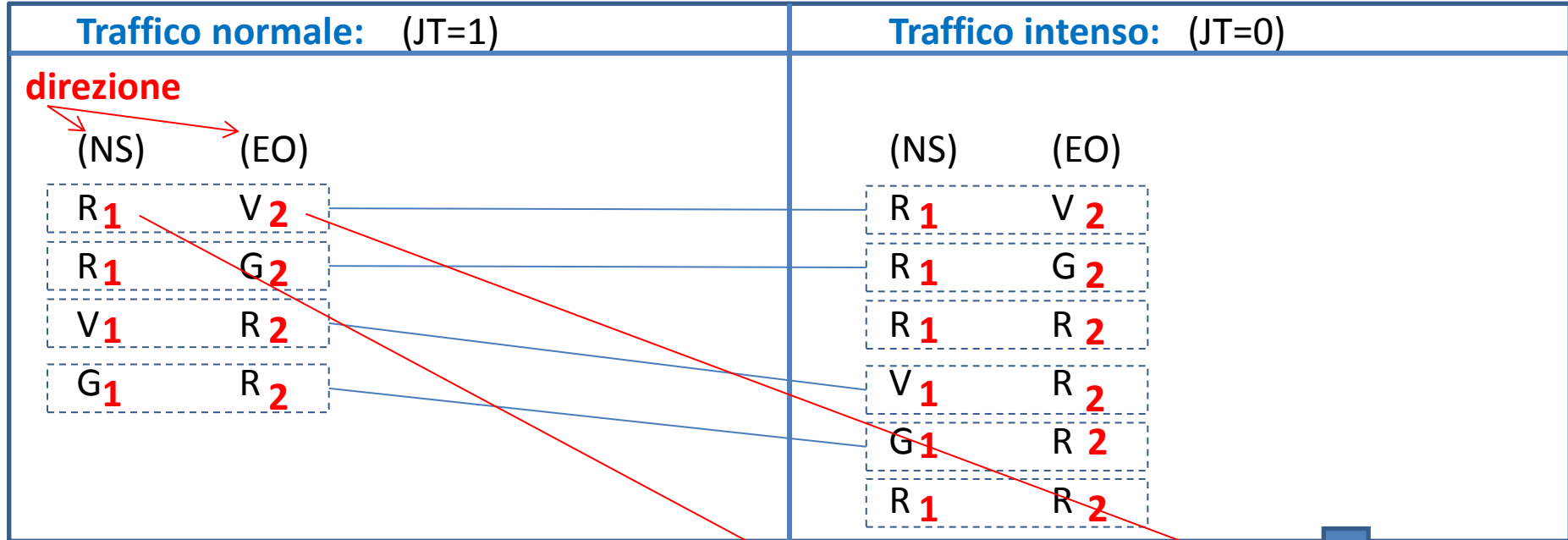


Per prima cosa dobbiamo decidere il numero minimo di stati che servono per descrivere il sistema. Quando JT=1 abbiamo 4 stati (pattern di luci), quando JT=0 abbiamo 6 stati ma non occorrono 10 stati (alcuni sono presenti sia nella configurazione a traffico normale che in quella a traffico intenso).

stato	R1	G1	V1	R2	G2	V2
1	1	0	0	0	0	1
2	1	0	0	0	1	0
3	1	0	0	1	0	0
4	0	0	1	1	0	0
5	0	1	0	1	0	0
6	1	0	0	1	0	0

# Esempio MSF : semaforo

scheduling

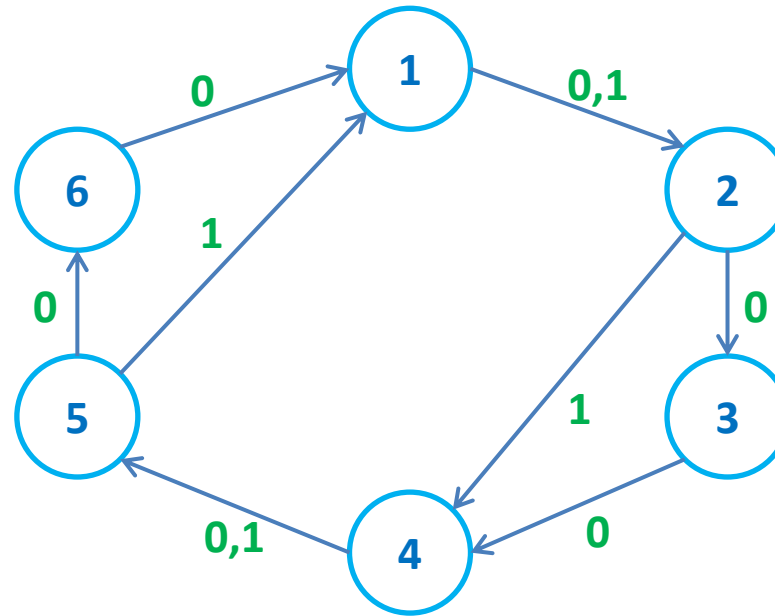


Per prima cosa dobbiamo decidere il numero minimo di stati che servono per descrivere il sistema. Quando JT=1 abbiamo 4 stati (pattern di luci), quando JT=0 abbiamo 6 stati ma non occorrono 10 stati (alcuni sono presenti sia nella configurazione a traffico normale che in quella a traffico intenso).

stato	R1	G1	V1	R2	G2	V2
1	1	0	0	0	0	1
2	1	0	0	0	1	0
3	1	0	0	1	0	0
4	0	0	1	1	0	0
5	0	1	0	1	0	0
6	1	0	0	1	0	0

# Esempio MSF : semaforo

JT



STG (ignoriamo output  
Per il momento)

## Osservazione I :

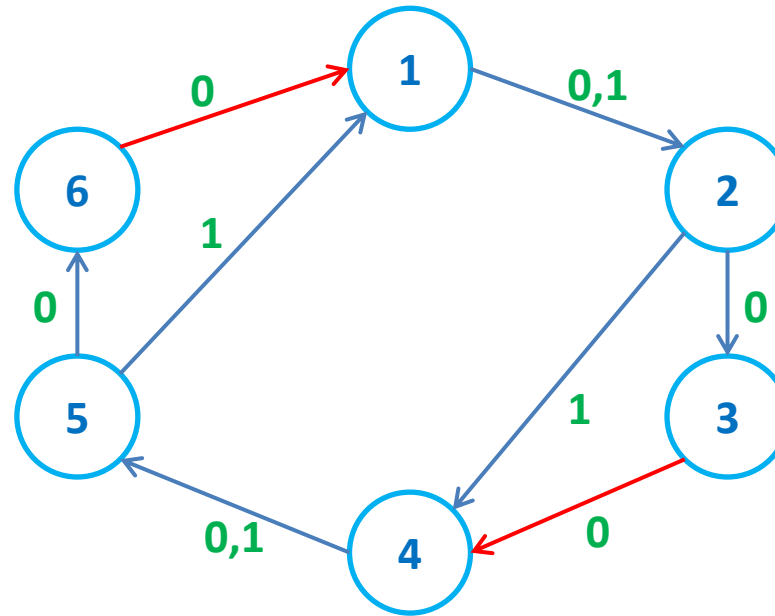
Due stati (3 e 6) hanno esattamente lo stesso pattern di luci. Possiamo fonderli?

**NO** → lo stato dopo il 3 è il 4 mentre lo stato dopo il 6 è lo stato 1!

stato	R1	G1	V1	R2	G2	V2
1	1	0	0	0	0	1
2	1	0	0	0	1	0
3	1	0	0	1	0	0
4	0	0	1	1	0	0
5	0	1	0	1	0	0
6	1	0	0	1	0	0

# Esempio MSF : semaforo

JT



**STG** (ignoriamo output  
Per il momento)

## Osservazione II :

STG ha struttura ciclica. Esistono due percorsi possibili. Se JT=1 (traffico moderato) gli stati 3 e 6 vengono saltati.

## Osservazione III:

A parte gli stati 3 e 6 ogni stato ha due possibili vie di uscita (a seconda del valore di JT)

stato	R1	G1	V1	R2	G2	V2
1	1	0	0	0	0	1
2	1	0	0	0	1	0
3	1	0	0	1	0	0
4	0	0	1	1	0	0
5	0	1	0	1	0	0
6	1	0	0	1	0	0



# Esempio MSF : semaforo

## **Selezione numero di flip-flop:**

Dato che il numero di stati è 6 il minimo numero di flip-flop che possiamo utilizzare per codificare gli stati è 3. Il numero massimo di flip-flop sarebbe 6 (uno per stato) ma sarebbe una scelta quanto meno discutibile. Con tre flip-flop e sei stati avremo 2 stati inutilizzati.

# Esempio MSF : semaforo

STT

Ci sono alcune regole euristiche per assegnare gli stati agli output dei flip-flop ma esse sono difficile da applicare e non garantiscono la minimizzazione del circuito. Una regola è quella di massimizzare il numero di 1 che compaiono nelle codifiche degli stati effettivamente utilizzati. L'idea è che un numero consistente di 1 possa rendere più facile la minimizzazione nelle mappe di Karnaugh. Su queste basi scegliamo di non utilizzare gli stati aventi codifica 000 e 001. Il resto delle combinazioni di output dei flip-flop sarà assegnata in ordine crescente durante la costruzione della tabella delle transizioni.

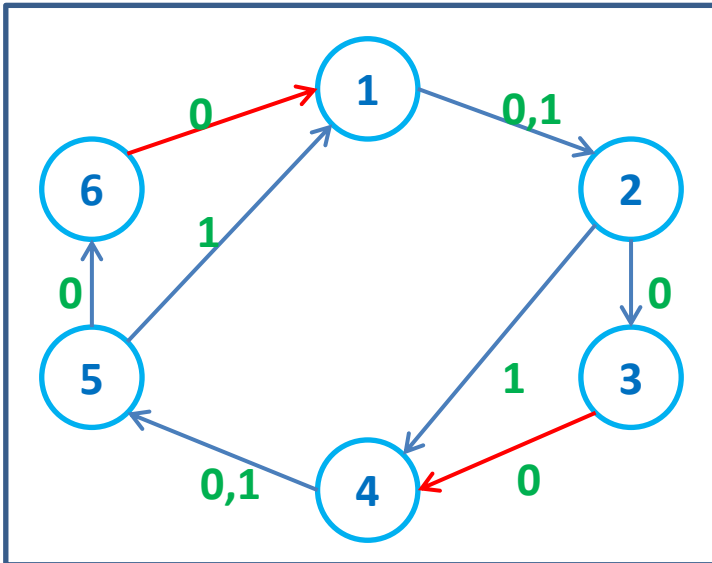
J	State (t)	Q1	Q2	Q3	State (t+1)	D1	D2	D3
0	<b>7</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
0	<b>8</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
0	1	0	1	0	2	0	1	1
0	2	0	1	1	3	1	0	0
0	3	1	0	0	4	1	0	1
0	4	1	0	1	5	1	1	0
0	5	1	1	0	6	1	1	1
0	6	1	1	1	1	0	1	0
1	<b>7</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
1	<b>8</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
1	1	0	1	0	2	0	1	1
1	2	0	1	1	4	1	0	1
1	3	1	0	0	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
1	4	1	0	1	5	1	1	0
1	5	1	1	0	1	0	1	0
1	6	1	1	1	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>

# Esempio MSF : semaforo

STT

J	State (t)	Q1	Q2	Q3	State (t+1)	D1	D2	D3
0	7	0	0	0	X	X	X	X
0	8	0	0	1	X	X	X	X
0	1	0	1	0	2	0	1	1
0	2	0	1	1	3	1	0	0
0	3	1	0	0	4	1	0	1
0	4	1	0	1	5	1	1	0
0	5	1	1	0	6	1	1	1
0	6	1	1	1	1	0	1	0
1	7	0	0	0	X	X	X	X
1	8	0	0	1	X	X	X	X
1	1	0	1	0	2	0	1	1
1	2	0	1	1	4	1	0	1
1	3	1	0	0	X	X	X	X
1	4	1	0	1	5	1	1	0
1	5	1	1	0	1	0	1	0
1	6	1	1	1	X	X	X	X

STG



# Esempio MSF : semaforo

STT

		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	1	0
	01	1	1	0	1
	11	X	1	X	0
	10	X	X	1	0

$$D1 = Q2' + Q3Q1' + J'Q1Q3'$$

J	State (t)	Q1	Q2	Q3	State (t+1)	D1	D2	D3
0	7	0	0	0	X	X	X	X
0	8	0	0	1	X	X	X	X
0	1	0	1	0	2	0	1	1
0	2	0	1	1	3	1	0	0
0	3	1	0	0	4	1	0	1
0	4	1	0	1	5	1	1	0
0	5	1	1	0	6	1	1	1
0	6	1	1	1	1	0	1	0
1	7	0	0	0	X	X	X	X
1	8	0	0	1	X	X	X	X
1	1	0	1	0	2	0	1	1
1	2	0	1	1	4	1	0	1
1	3	1	0	0	X	X	X	X
1	4	1	0	1	5	1	1	0
1	5	1	1	0	1	0	1	0
1	6	1	1	1	X	X	X	X

# Esempio MSF : semaforo

STT

		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	0	1
	01	0	1	1	1
	11	X	1	X	1
	10	X	X	0	1

$$D2 = Q1Q3 + Q2Q3'$$

J	State (t)	Q1	Q2	Q3	State (t+1)	D1	D2	D3
0	7	0	0	0	X	X	X	X
0	8	0	0	1	X	X	X	X
0	1	0	1	0	2	0	1	1
0	2	0	1	1	3	1	0	0
0	3	1	0	0	4	1	0	1
0	4	1	0	1	5	1	1	0
0	5	1	1	0	6	1	1	1
0	6	1	1	1	1	0	1	0
1	7	0	0	0	X	X	X	X
1	8	0	0	1	X	X	X	X
1	1	0	1	0	2	0	1	1
1	2	0	1	1	4	1	0	1
1	3	1	0	0	X	X	X	X
1	4	1	0	1	5	1	1	0
1	5	1	1	0	1	0	1	0
1	6	1	1	1	X	X	X	X

# Esempio MSF : semaforo

STT

		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	0	1
	01	1	0	0	1
	11	X	0	X	0
	10	X	X	1	1

$$D3 = J'Q3' + J Q1'$$

J	State (t)	Q1	Q2	Q3	State (t+1)	D1	D2	D3
0	7	0	0	0	X	X	X	X
0	8	0	0	1	X	X	X	X
0	1	0	1	0	2	0	1	1
0	2	0	1	1	3	1	0	0
0	3	1	0	0	4	1	0	1
0	4	1	0	1	5	1	1	0
0	5	1	1	0	6	1	1	1
0	6	1	1	1	1	0	1	0
1	7	0	0	0	X	X	X	X
1	8	0	0	1	X	X	X	X
1	1	0	1	0	2	0	1	1
1	2	0	1	1	4	1	0	1
1	3	1	0	0	X	X	X	X
1	4	1	0	1	5	1	1	0
1	5	1	1	0	1	0	1	0
1	6	1	1	1	X	X	X	X

# Esempio MSF : semaforo

		Q2 Q3			
		00	01	11	10
J Q1	00	1	1	1	0
	01	1	1	0	1
	11	1	1	0	0
	10	1	1	1	0

		Q2 Q3			
		00	01	11	10
J Q1	00	0	0	0	1
	01	0	1	1	1
	11	0	1	1	1
	10	0	0	0	1

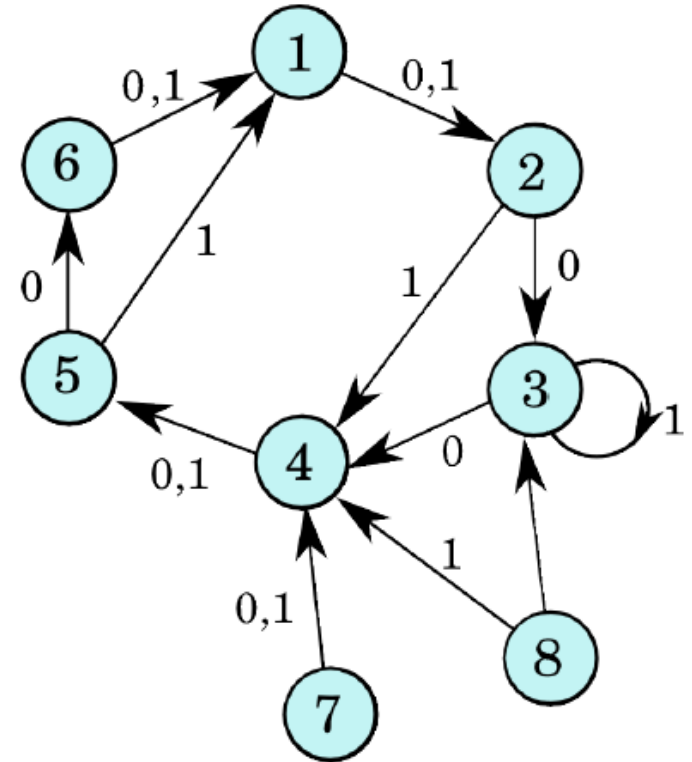
		Q2 Q3			
		00	01	11	10
J Q1	00	1	0	0	1
	01	1	0	0	1
	11	0	0	0	0
	10	1	1	1	1

La definizione del circuito sequenziale è completa. Ora dobbiamo verificare se il circuito si comporta in modo coerente anche se il suo ciclo di funzionamento parte da uno degli stati inutilizzati.

Un modo conveniente di fare questo test è quello di ridisegnare STG includendo gli stati non utilizzati (ora, grazie alle mappe, abbiamo anche i valori da associare agli stati inizialmente non utilizzati). Possiamo quindi modificare la STT e da essa ricavare la nuova versione di STG.

# Esempio MSF : semaforo

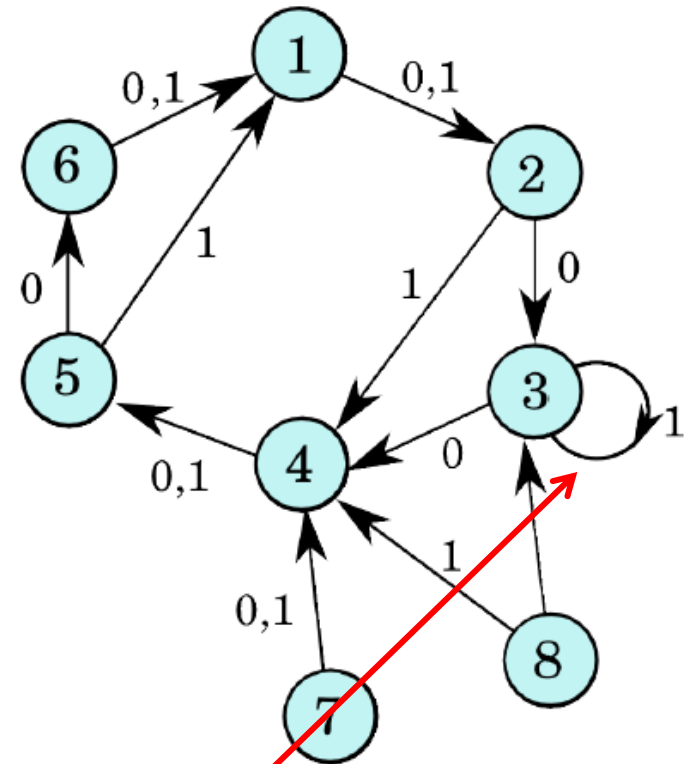
J	State(t)	Q1	Q2	Q3	State(t+1)	D1	D2	D3
0	7	0	0	0	4	1	0	1
0	8	0	0	1	3	1	0	0
0	1	0	1	0	2	0	1	1
0	2	0	1	1	3	1	0	0
0	3	1	0	0	4	1	0	1
0	4	1	0	1	5	1	1	0
0	5	1	1	0	6	1	1	1
0	6	1	1	1	1	0	1	0
1	7	0	0	0	4	1	0	1
1	8	0	0	1	4	1	0	1
1	1	0	1	0	2	0	1	1
1	2	0	1	1	4	1	0	1
1	3	1	0	0	3	1	0	0
1	4	1	0	1	5	1	1	0
1	5	1	1	0	1	0	1	0
1	6	1	1	1	1	0	1	0





# Esempio MSF : semaforo

J	State(t)	Q1	Q2	Q3	State(t+1)	D1	D2	D3
0	7	0	0	0	4	1	0	1
0	8	0	0	1	3	1	0	0
0	1	0	1	0	2	0	1	1
0	2	0	1	1	3	1	0	0
0	3	1	0	0	4	1	0	1
0	4	1	0	1	5	1	1	0
0	5	1	1	0	6	1	1	1
0	6	1	1	1	1	0	1	0
1	7	0	0	0	4	1	0	1
1	8	0	0	1	4	1	0	1
1	1	0	1	0	2	0	1	1
1	2	0	1	1	4	1	0	1
1	3	1	0	0	3	1	0	0
1	4	1	0	1	5	1	1	0
1	5	1	1	0	1	0	1	0
1	6	1	1	1	1	0	1	0

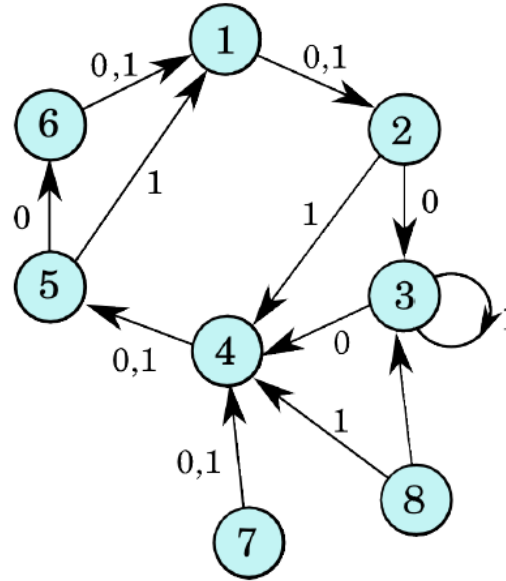


## PROBLEMA:

Se il valore di J è 1 (incrocio a traffico moderato) e il circuito parte in stato 3 RIMANE INTRAPPOLATO in questo stato senza poter uscire da esso.

# Esempio MSF : semaforo

J	State(t)	Q1	Q2	Q3	State(t+1)	D1	D2	D3
0	7	0	0	0	4	1	0	1
0	8	0	0	1	3	1	0	0
0	1	0	1	0	2	0	1	1
0	2	0	1	1	3	1	0	0
0	3	1	0	0	4	1	0	1
0	4	1	0	1	5	1	1	0
0	5	1	1	0	6	1	1	1
0	6	1	1	1	1	0	1	0
1	7	0	0	0	4	1	0	1
1	8	0	0	1	4	1	0	1
1	1	0	1	0	2	0	1	1
1	2	0	1	1	4	1	0	1
1	3	1	0	0	3	1	0	0
1	4	1	0	1	5	1	1	0
1	5	1	1	0	1	0	1	0
1	6	1	1	1	1	0	1	0



**Rimaniamo bloccati nello stato 3 se : a) J=1 e passiamo allo stato 3 dallo stato 2 b) il circuito si trova in stato 3 quando cambiamo il valore di J da 0 a 1.**

Il problema si verifica nella cella 1100 della mappa di Karnaugh. Guardando le tre mappe troviamo che possiamo risolvere il problema nella mappa di D3, cambiando uno 0 in 1. In questo modo se sono in stato 3 e J=1 andrò in stato 4 invece di restare bloccato in stato 3.

		Q2 Q3			
		00	01	11	10
J Q1	00	1	0	0	1
	01	1	0	0	1
	11	0	0	0	0
	10	1	1	1	1



		Q2 Q3			
		00	01	11	10
J Q1	00	1	0	0	1
	01	1	0	0	1
	11	1	0	0	0
	10	1	1	1	1

$$D3 = J'Q3' + JQ1' + Q2'Q3'$$

# Esempio MSF : semaforo

## Sintesi circuiti di output:

Finora ci siamo occupati solo delle transizioni tra stati ignorando i circuiti che, per ogni stato, generano i segnali che determinano l'accensione del corretto pattern di luci. Esistono sei di questi circuiti ma essi dipendono tutti da tre segnali di input: Q1, Q2 e Q3.

Le mappe di Karnaugh dei circuiti possono essere riempite sulla base dei pattern di luci definiti inizialmente per ogni stato. Anche in questo caso inizieremo ignorando gli stati non utilizzati.

Come visto in precedenza i valori X nelle mappe verranno rimpiazzati, dopo la minimizzazione, da valori 1 o 0 e questo fornirà i pattern di luci emessi dal circuito negli stati 7 e 8. Come abbiamo fatto in precedenza dovremo verificare se i pattern di luci degli stati inutilizzati sono sicuri o se possono causare problemi.

# Esempio MSF : semaforo

## Sintesi circuiti di output:

Le mappe di Karnaugh vengono riempite a partire dai pattern acceso/spento presenti nelle specifiche iniziali.

State	Q1	Q2	Q3	R1	G1	V1	R2	G2	V2
1	0	1	0	1	0	0	0	0	1
2	0	1	1	1	0	0	0	1	0
3	1	0	0	1	0	0	1	0	0
4	1	0	1	0	0	1	1	0	0
5	1	1	0	0	1	0	1	0	0
6	1	1	1	1	0	0	1	0	0
7	0	0	0	X	X	X	X	X	X
8	0	0	1	X	X	X	X	X	X

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	1
	1	1	0	1	0

$$R1 = Q1' + Q2'Q3' + Q2Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	0	0	1

$$G1 = Q1Q2Q3'$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	1	0	0

$$V1 = Q2'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	1	1	1	1

$$R2 = Q1$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	0
	1	0	0	0	0

$$G2 = Q1'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	1
	1	0	0	0	0

$$V2 = Q1'Q3'$$

# Esempio MSF : semaforo

Test stati inutilizzati:

**Stato 7** ha valori Q 000.

Nelle mappe per R1 e V2 le X nella cella che corrisponde a Q 000 verranno impostate a 1, nelle altre mappe saranno impostate a 0.

Quindi se il sistema dovesse partire in stato 7 le luci sarebbero Rosso in una direzione e Verde nell'altra.

Nessun problema.

State	Q1	Q2	Q3	R1	G1	V1	R2	G2	V2
1	0	1	0	1	0	0	0	0	1
2	0	1	1	1	0	0	0	1	0
3	1	0	0	1	0	0	1	0	0
4	1	0	1	0	0	1	1	0	0
5	1	1	0	0	1	0	1	0	0
6	1	1	1	1	0	0	1	0	0
7	0	0	0	X	X	X	X	X	X
8	0	0	1	X	X	X	X	X	X

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	1
	1	1	0	1	0

$$R1 = Q1' + Q2'Q3' + Q2Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	0	0	1

$$G1 = Q1Q2Q3'$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	1	0	0

$$V1 = Q2'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	1	1	1	1

$$R2 = Q1$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	0
	1	0	0	0	0

$$G2 = Q1'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	1
	1	0	0	0	0

$$V2 = Q1'Q3'$$

# Esempio MSF : semaforo

Test stati inutilizzati:

**Stato 8** ha valori Q 001.

Se partiamo in questo stato guardando il contenuto delle mappe vediamo che R1, V1 e G2 saranno impostati a 1 e tutte le altre luci saranno impostate a 0. Avremo le luci rosso e verde accese in una direzione e solo il giallo nell'altra.

Potrebbe essere un problema ma questa situazione si verifica solo per uno stato (se siamo in 8 all'accensione) e poi si esce da stato 8 e tutto funziona in modo regolare.

State	Q1	Q2	Q3	R1	G1	V1	R2	G2	V2
1	0	1	0	1	0	0	0	0	1
2	0	1	1	1	0	0	0	1	0
3	1	0	0	1	0	0	1	0	0
4	1	0	1	0	0	1	1	0	0
5	1	1	0	0	1	0	1	0	0
6	1	1	1	1	0	0	1	0	0
7	0	0	0	X	X	X	X	X	X
8	0	0	1	X	X	X	X	X	X

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	1
	1	1	0	1	0

$$R1 = Q1' + Q2'Q3' + Q2Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	0	0	1

$$G1 = Q1Q2Q3'$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	1	0	0

$$V1 = Q2'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	1	1	1	1

$$R2 = Q1$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	0
	1	0	0	0	0

$$G2 = Q1'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	1
	1	0	0	0	0

$$V2 = Q1'Q3'$$

# Esempio MSF : semaforo

In definitiva i circuiti da costruire sono i seguenti:

State	Q1	Q2	Q3	R1	G1	V1	R2	G2	V2
1	0	1	0	1	0	0	0	0	1
2	0	1	1	1	0	0	0	1	0
3	1	0	0	1	0	0	1	0	0
4	1	0	1	0	0	1	1	0	0
5	1	1	0	0	1	0	1	0	0
6	1	1	1	1	0	0	1	0	0
7	0	0	0	X	X	X	X	X	X
8	0	0	1	X	X	X	X	X	X

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	1
	1	1	0	1	0

$$R1 = Q1' + Q2'Q3' + Q2Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	0	0	1

$$G1 = Q1Q2Q3'$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	1	0	0

$$V1 = Q2'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	1	1	1	1

$$R2 = Q1$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	0
	1	0	0	0	0

$$G2 = Q1'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	1
	1	0	0	0	0

$$V2 = Q1'Q3'$$

$$D1 = Q2' + J' \cdot Q1 \cdot Q3' + Q1' \cdot Q3 \quad D2 = Q1 \cdot Q3 + Q2 \cdot Q3' \quad D3 = Q2' \cdot Q3' + J' \cdot Q3' + J \cdot Q1'$$

$$R1 = Q1' + Q2' \cdot Q3' + Q2 \cdot Q3$$

$$G1 = Q2 \cdot Q1 \cdot Q3'$$

$$V1 = Q2' \cdot Q3$$

$$R2 = Q1$$

$$G2 = Q1' \cdot Q3$$

$$V2 = Q1' \cdot Q3'$$

I termini comuni sono sottolineati ... vanno implementati **una sola volta** in hardware.

# Esempio MSF : semaforo

## Variante di assegnamento degli stati:

Se volessimo provare a semplificare il circuito potremmo provare altri schemi di assegnamento dei flip-flop per gli stati. Un possibile punto di partenza potrebbe essere quello di provare a semplificare il circuito di output (quello che governa il pattern delle luci).

In esso potremmo imporre, ad esempio, che  $R1=Q1$  e  $R2=Q2$ . Il terzo output,  $Q3$ , va assegnato in modo che tutti gli stati risultino distinti. Un possibile set di assegnamenti è riportato di seguito.

State	Q1	Q2	Q3	R1	G1	V1	R2	G2	V2
1	1	0	0	1	0	0	0	0	1
2	1	0	1	1	0	0	0	1	0
3	1	1	1	1	0	0	1	0	0
4	0	1	1	0	0	1	1	0	0
5	0	1	0	0	1	0	1	0	0
6	1	1	0	1	0	0	1	0	0
7	0	0	0	X	X	X	X	X	X
8	0	0	1	X	X	X	X	X	X



# Esempio MSF : semaforo

Variante di assegnamento degli stati:

State	Q1	Q2	Q3	R1	G1	V1	R2	G2	V2
1	1	0	0	1	0	0	0	0	1
2	1	0	1	1	0	0	0	1	0
3	1	1	1	1	0	0	1	0	0
4	0	1	1	0	0	1	1	0	0
5	0	1	0	0	1	0	1	0	0
6	1	1	0	1	0	0	1	0	0
7	0	0	0	X	X	X	X	X	X
8	0	0	1	X	X	X	X	X	X

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	1	1	1	1

$$R1 = Q1$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	1
	1	0	0	0	0

$$G1 = Q1'Q3'$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	0
	1	0	0	0	0

$$V1 = Q1'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	1	1
	1	0	0	1	1

$$R2 = Q2$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	0	1	0	0

$$G2 = Q2'Q3$$

		Q2 Q3			
		00	01	11	10
Q1	0	X	X	0	0
	1	1	0	0	0

$$V2 = Q2'Q3'$$

I circuiti di output sono più semplici ma dobbiamo ridefinire anche la logica circuitale di transizione tra gli stati ...

# Esempio MSF : semaforo

Variante di assegnamento degli stati:

J	State(t)	Q1	Q2	Q3	State(t+1)	D1	D2	D3
0	7	0	0	0	X	X	X	X
0	8	0	0	1	X	X	X	X
0	1	1	0	0	2	1	0	1
0	2	1	0	1	3	1	1	1
0	3	1	1	1	4	0	1	1
0	4	0	1	1	5	0	1	0
0	5	0	1	0	6	1	1	0
0	6	1	1	0	0	1	0	0
1	7	0	0	0	X	X	X	X
1	8	0	0	1	X	X	X	X
1	1	1	0	0	2	1	0	1
1	2	1	0	1	4	0	1	1
1	3	1	1	1	X	X	X	X
1	4	0	1	1	5	0	1	0
1	5	0	1	0	1	1	0	0
1	6	1	1	0	X	X	X	X

		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	0	1
	01	1	1	0	1
	11	1	0	X	X
	10	X	X	0	1

$$D1 = J'Q2' + Q3'$$

		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	1	1
	01	0	1	1	0
	11	0	1	X	X
	10	X	X	1	0

$$D2 = J'Q1' + Q3$$

		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	0	0
	01	1	1	1	0
	11	1	1	X	X
	10	X	X	0	0

$$D3 = Q2' + Q1Q3$$

Il circuito risultante è più semplice del precedente, e se verifichiamo il comportamento degli stati inutilizzati non ci sono problemi (verificate).

# Esempio MSF : semaforo

Variante di assegnamento degli stati:

J	State(t)	Q1	Q2	Q3	State(t+1)	D1	D2	D3
0	7	0	0	0	X	X	X	X
0	8	0	0	1	X	X	X	X
0	1	1	0	0	2	1	0	1
0	2	1	0	1	3	1	1	1
0	3	1	1	1	4	0	1	1
0	4	0	1	1	5	0	1	0
0	5	0	1	0	6	1	1	0
0	6	1	1	0	0	1	0	0
1	7	0	0	0	X	X	X	X
1	8	0	0	1	X	X	X	X
1	1	1	0	0	2	1	0	1
1	2	1	0	1	4	0	1	1
1	3	1	1	1	X	X	X	X
1	4	0	1	1	5	0	1	0
1	5	0	1	0	1	1	0	0
1	6	1	1	0	X	X	X	X

		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	0	1
	01	1	1	0	1
	11	1	0	X	X
	10	X	X	0	1

$$D1 = J'Q2' + Q3'$$

		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	1	1
	01	0	1	1	0
	11	0	1	X	X
	10	X	X	1	0

$$D2 = J'Q1' + Q3$$

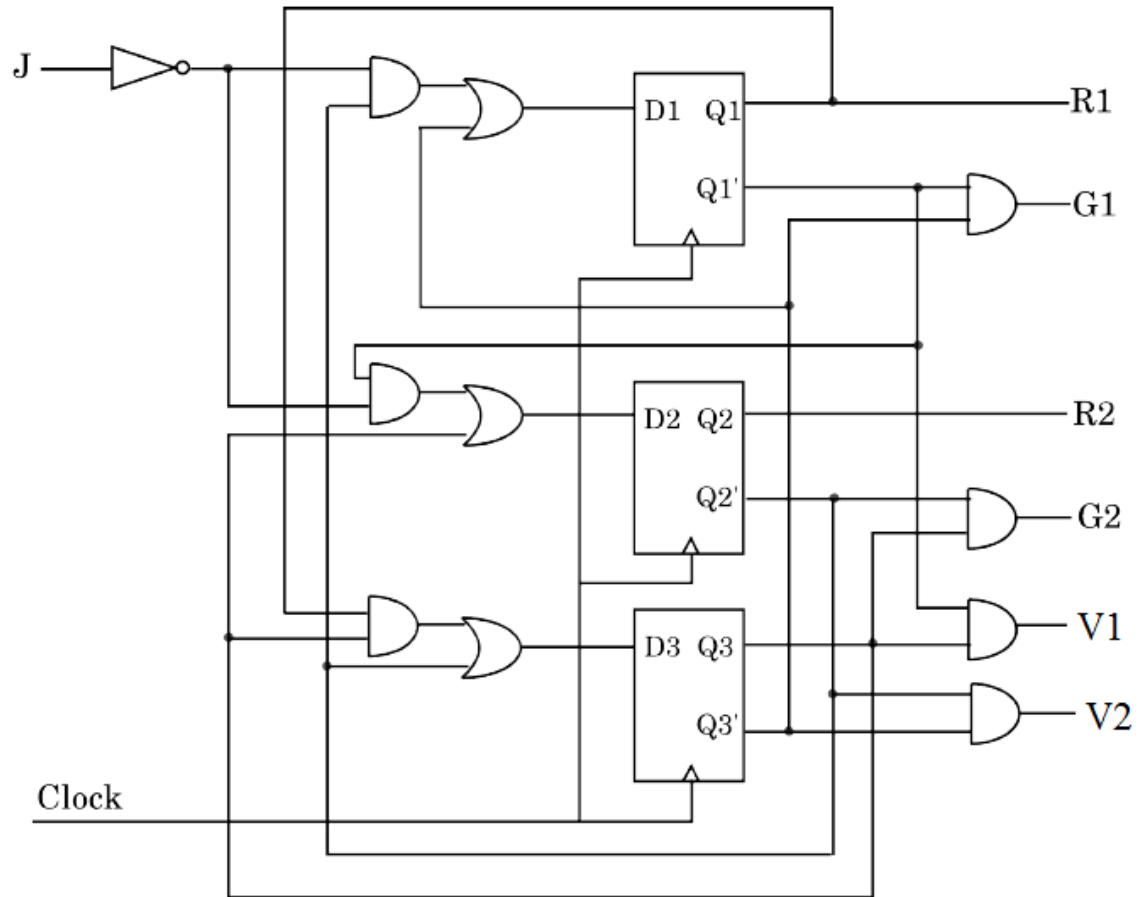
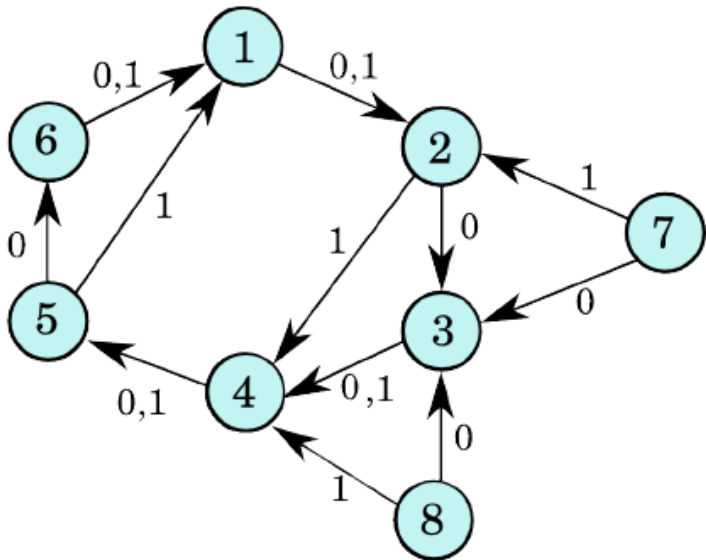
		Q2 Q3			
		00	01	11	10
J Q1	00	X	X	0	0
	01	1	1	1	0
	11	1	1	X	X
	10	X	X	0	0

$$D3 = Q2' + Q1Q3$$

Il circuito risultante è più semplice del precedente, e se verifichiamo il comportamento degli stati inutilizzati non ci sono problemi (verificate).

# Esempio MSF : semaforo

STG finale e circuito :



Implementiamo il circuito in Logisim