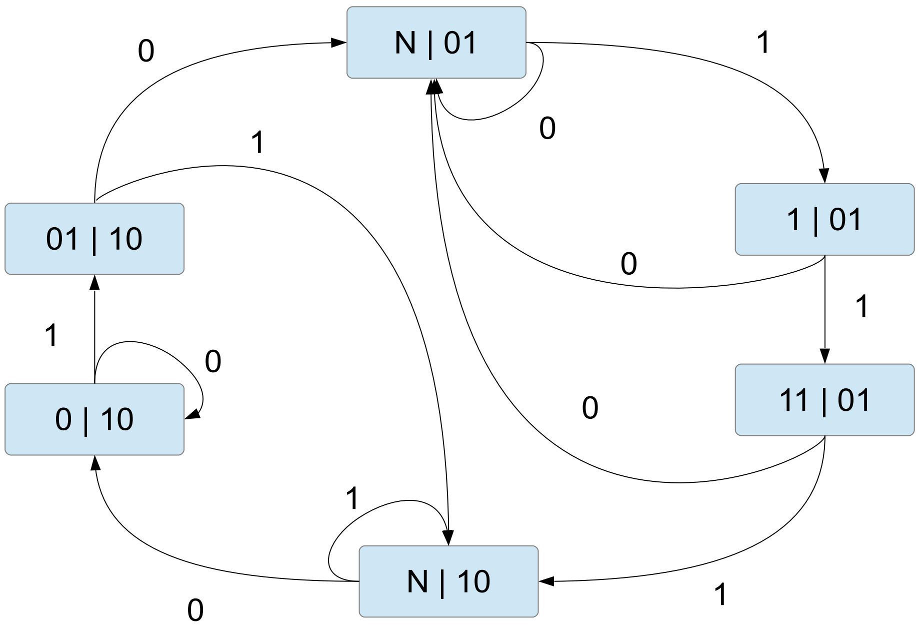
L'automa dato ha un comportamento simile ad un Flip-Flop Toogle. L'alfabeto di uscita è O={**10**,**01**}. La transazione da **01** a **10** è ottenuta presentando in ingresso la sequenza **111** mentre la transizione opposta da **10** a **01** è ottenuta con la sequenza **010**. L'automa deve ricordare quale uscita sta di volta in volta presentando in uscita, in modo da conservarla fino a che non viene riconosciuta una parola utile. In aggiunta, a seconda dell'uscita corrente l'automa dovrà riconoscere una sola delle due configurazioni date. Quando l'uscita **01** l'automa deve attendere che compaia in ingresso la sequenza **111** per passare all'altra configurazione di uscita mentre la sequenza **010** è ininfluente. In maniera analoga quando le uscite sono **10** occorre la sequenza **010** per effettuare la transizione in uscita mentre la sequenza **111** è in questo caso ininfluente.
Assumiamo che l'automa non abbia conoscenza precedente del flusso di dati in arrivo, cioè nessun prefisso utile può essere usato per costruire la successiva parola. L'uscita iniziale è **01**; in questo caso l'automa si comporta come un riconoscitore della stringa **111**. Quando l'uscita passa a **10**, l'automa diventa un riconoscitore della stringa **010**.

Lo State Transition Graph è il seguente:



La corrispondente tabella STT è la seguente:

State	**I=0**	**I=1**	**O**
N|01	N|01	1|01	01
1|01	N|01	11|01	01
11|01	N|01	N|10	01
N|10	0|10	N|10	10
0|10	0|10	01|10	10
01|10	N|01	N|10	10

Per codificare i 6 stati possibili occorrono ceil($\log_2(6)$)=3 bit. Codifichiamo gli stati usando un bit per memorizzare l'uscita corrente e due bit per codificare il numero di caratteri riconosciuti:

State	**$S_2S_1S_0$**
N|01	000
1|01	010
11|01	100
N|10	001
0|10	011
01|10	101

Codifichiamo l'alfabeto di ingresso e l'alfabeto di uscita in maniera triviale:

I	**I_0**
0	0
1	1

Codifichiamo l'alfabeto di ingresso e l'alfabeto di uscita in maniera triviale:

O	**O_1O_0**
01	01
10	10

La STT codificata diventa:

$S_2S_1S_0$	**I=0**	**I=1**	**O_1O_0**
000	000	010	01
010	000	100	01
100	000	001	01
001	011	001	10
011	011	101	10
101	000	001	10

La funzione Lambda di uscita è banalmente:

$$\mathbf{O_0 = \sim S_0}$$
$$\mathbf{O_1 = S_0}$$

Cammino critico e complessità uguale a 0.

Applichiamo Karnaugh per ogni bit della funzione Delta:

$S_0 I$	00	01	11	10
$S_2 S_1$				
00	000	010	001	011
01	000	100	101	011
11	xxx	xxx	xxx	xxx
10	000	001	001	000

S_2^* $S_0 I$	00	01	11	10
$S_2 S_1$				
00	0	0	0	0
01	0	1	1	0
11	x=0	x=1	x=1	x=0
10	0	0	0	0

$$S_2^* = S_1 I$$

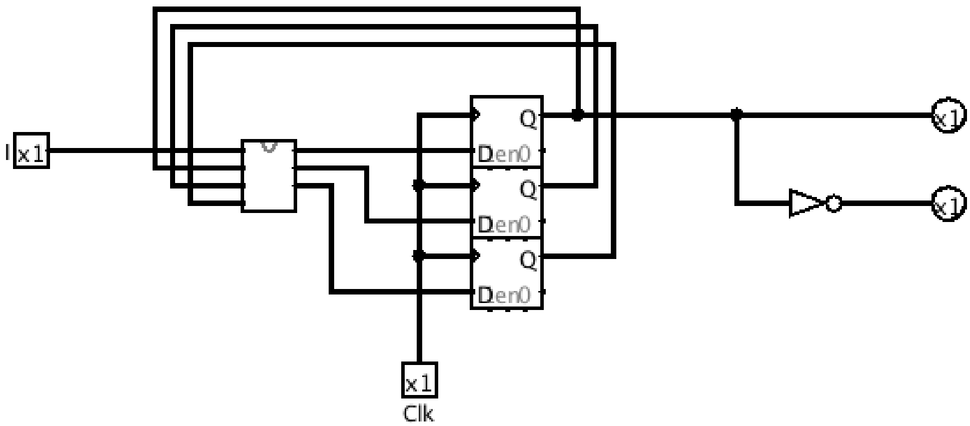
S_1^* $S_0 I$	00	01	11	10
$S_2 S_1$				
00	0	1	0	1
01	0	0	0	1
11	x=0	x=0	x=0	x=0
10	0	0	0	0

$$S_1^* = \sim S_2 \sim S_1 \sim S_0 I + \sim S_2 S_0 \sim I$$

S_0^* S_0I	00	01	11	10
S_2S_1				
00	0	0	1	1
01	0	0	1	1
11	x=0	x=1	x=1	x=0
10	0	1	1	0

$$S_0^* = S_2 I + \sim S_2 S_0$$

Il circuito dell'automa di Huffman corrispondente è il seguente:



Il curcuito della funzione delta è il seguente: