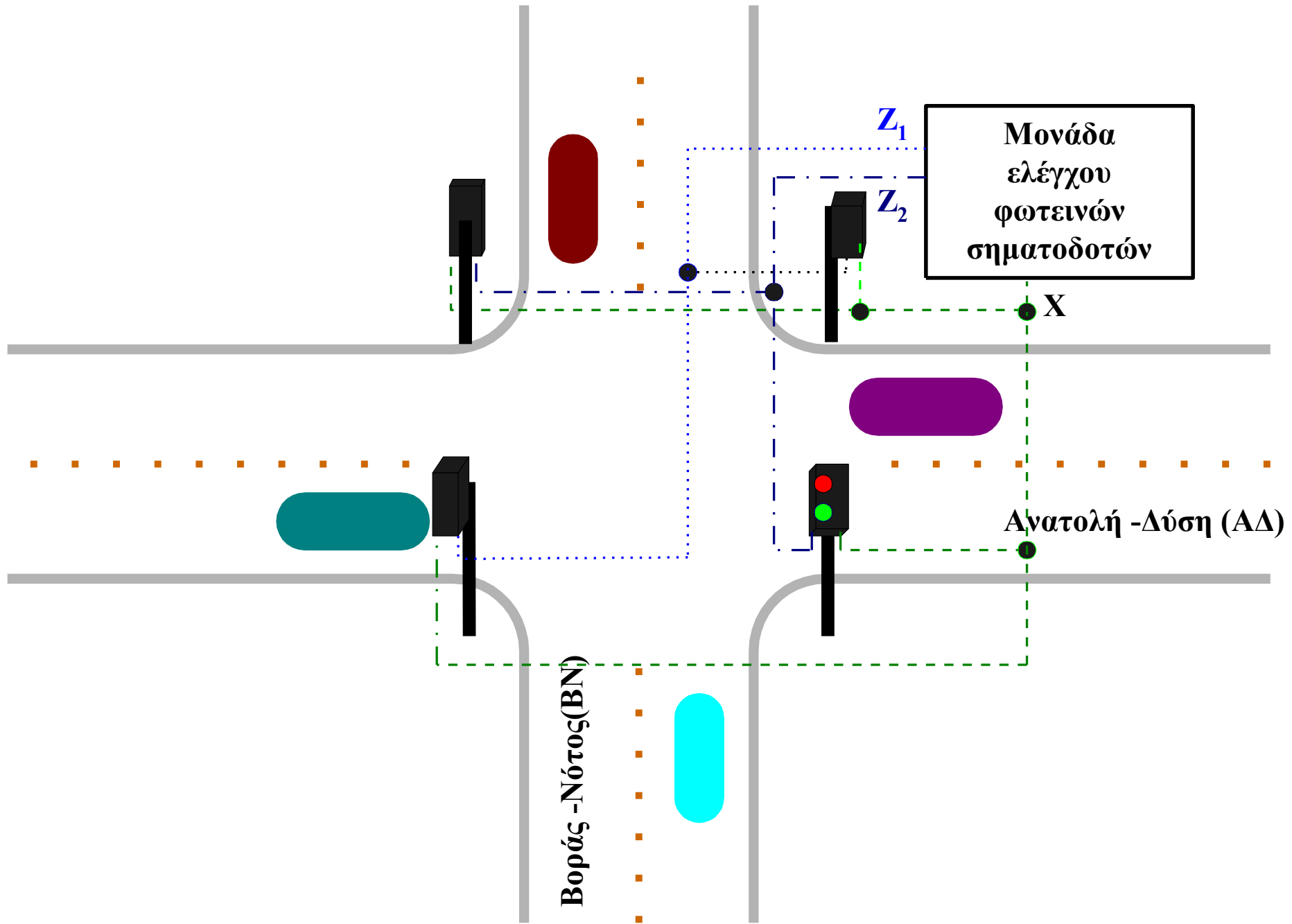


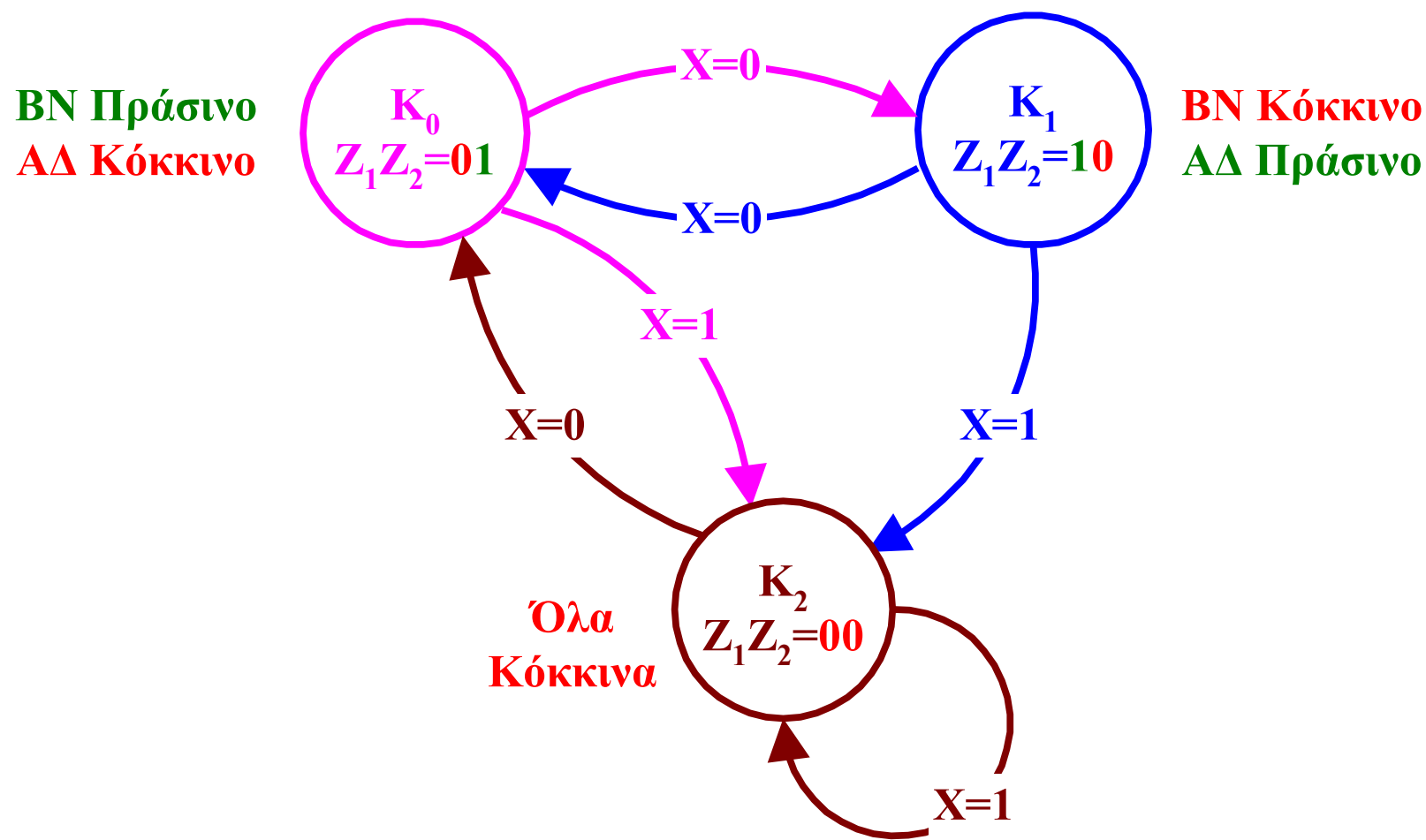
# Παράδειγμα Ακολουθιακού Κυκλώματος

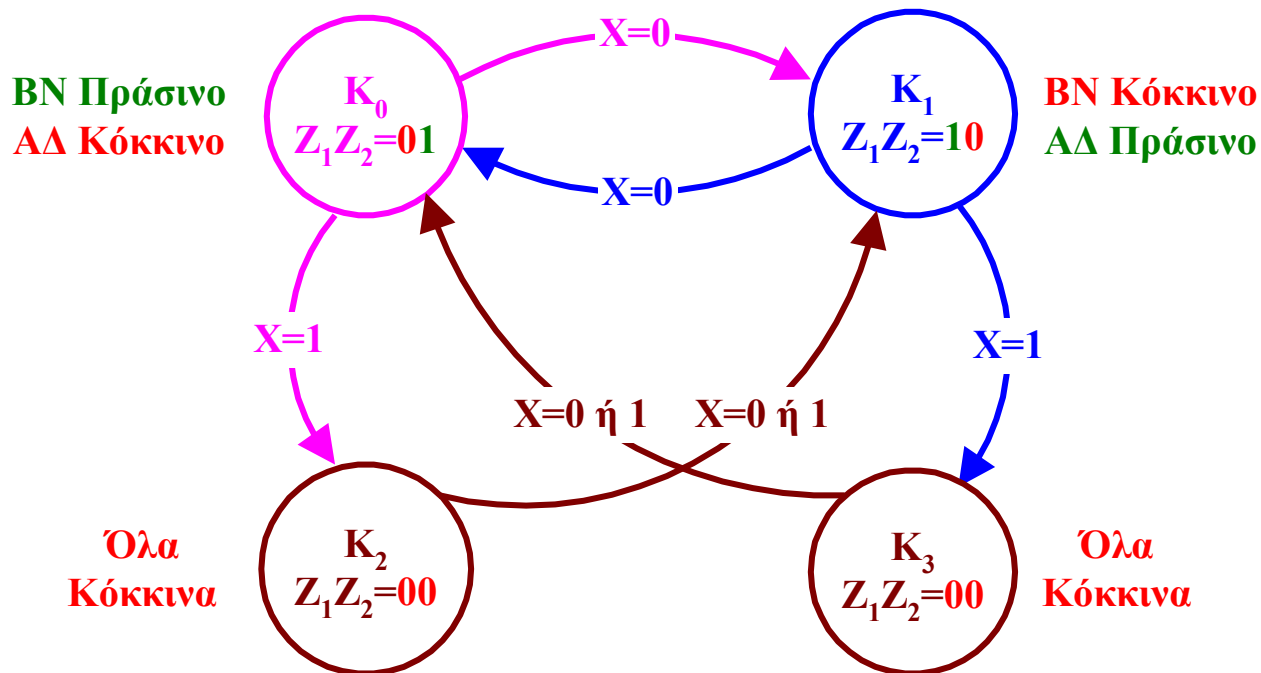
Να σχεδιαστεί ένα κύκλωμα ελέγχου που θα ελέγχει τους φωτεινούς σηματοδότες κυκλοφορίας σε μία διασταύρωση δευτερευόντων δρόμων, όπως αυτή που φαίνεται στο σχήμα.

Οι είσοδοι στο κύκλωμα ελέγχου είναι σήματα που προέρχονται από διακόπτες (push button), που ενεργοποιούνται από πεζούς που θέλουν να διασχίσουν το δρόμο. Όταν δεν υπάρχουν πεζοί οι σηματοδότες αφήνουν να κυκλοφορήσουν τα αυτοκίνητα ανά ένα λεπτό σε κάθε δρόμο. Αυτό σημαίνει, ότι, όταν τα αυτοκίνητα θα κινούνται στο δρόμο ΑΔ (Ανατολή-Δύση), οι αντίστοιχοι σηματοδότες θα είναι πράσινοι και οι σηματοδότες του δρόμου ΒΝ (Βοράς-Νότος) κόκκινοι, και όταν τα αυτοκίνητα θα κινούνται στο δρόμο ΒΝ, οι αντίστοιχοι σηματοδότες θα είναι πράσινοι και οι σηματοδότες του δρόμου ΑΔ κόκκινοι κ.ο.κ. Αν σε κάποια στιγμή ενεργοποιηθεί ο διακόπτης από κάποιον πεζό, τότε όλοι οι σηματοδότες θα γίνουν κόκκινοι για ένα λεπτό επιτρέποντας στον πεζό να διασχίσει οποιονδήποτε από τους δύο δρόμους. Για να κρατηθεί το πρόβλημα μικρό, θα αγνοηθεί η πορτοκαλί κατάσταση των σηματοδοτών.



Με βάση την εκφώνηση του προβλήματος κατασκευάζεται το διάγραμμα καταστάσεων μοντέλου Moore.





Απ' το διάγραμμα καταστάσεων κατασκευάζεται ο πίνακας καταστάσεων.

Πίνακας καταστάσεων (μοντέλου Moore)

Παρούσα κατάσταση	Είσοδος		Έξοδος
	$X=0$	$X=1$	
$Q_1Q_2$			$Z_1Z_2$
$K_0$	$K_1$	$K_2$	0 1
$K_1$	$K_0$	$K_3$	1 0
$K_2$	$K_1$	$K_1$	0 0
$K_3$	$K_0$	$K_0$	0 0

Επόμενη κατάσταση

Πίνακας καταστάσεων (μοντέλου Moore)

Παρούσα κατάσταση	Είσοδος		Έξοδος
	X=0	X=1	
Q <sub>1</sub> Q <sub>2</sub>			Z <sub>1</sub> Z <sub>2</sub>
K <sub>0</sub>	K <sub>1</sub>	K <sub>2</sub>	0 1
K <sub>1</sub>	K <sub>0</sub>	K <sub>3</sub>	1 0
K <sub>2</sub>	K <sub>1</sub>	K <sub>1</sub>	0 0
K <sub>3</sub>	K <sub>0</sub>	K <sub>0</sub>	0 0

Επόμενη κατάσταση

Γίνεται η παρακάτω τυχαία εκχώρηση καταστάσεων:  $Q_1Q_2=00=K_0$   $Q_1Q_2=01=K_1$   $Q_1Q_2=10=K_2$   
 $Q_1Q_2=11=K_3$

Πίνακας καταστάσεων

Παρούσα κατάσταση	Είσοδος		Έξοδος
	X=0	X=1	
Q <sub>1</sub> Q <sub>2</sub>			Z <sub>1</sub> Z <sub>2</sub>
00	01	10	0 1
01	00	11	1 0
10	01	01	0 0
11	00	00	0 0

Επόμενη κατάσταση

Πίνακας καταστάσεων

Παρούσα κατάσταση	Είσοδος		Έξοδος
$Q_1Q_2$	$X=0$	$X=1$	$Z_1Z_2$
00	01	10	0 1
01	00	11	1 0
10	01	01	0 0
11	00	00	0 0

Επόμενη κατάσταση

Πίνακας μεταβάσεων

Πρωτεύουσα είσοδος	Παρούσα κατάσταση	Επόμενη κατάσταση	Πρωτεύουσες έξοδοι
$X$	$Q_1Q_2$	$Q_1^+Q_2^+$	$Z_1Z_2$
0	00	01	0 1
0	01	00	1 0
0	10	01	0 0
0	11	00	0 0
1	00	10	0 1
1	01	11	1 0
1	10	01	0 0
1	11	00	0 0

Επιλέγεται D flip-flop και από τον πίνακα καταστάσεων κατασκευάζεται ο πίνακας διεγέρσεων:

Πίνακας διεγέρσεων				
Πρωτεύουσα είσοδος	Παρούσα κατάσταση	Επόμενη κατάσταση	Δευτερεύουσες έξοδοι	Πρωτεύουσες έξοδοι
X	Q <sub>1</sub> Q <sub>2</sub>	Q <sub>1</sub> <sup>+</sup> Q <sub>2</sub> <sup>+</sup>	D <sub>1</sub> D <sub>2</sub>	Z <sub>1</sub> Z <sub>2</sub>
0	0 0	0 1	0 1	0 1
0	0 1	0 0	0 0	1 0
0	1 0	0 1	0 1	0 0
0	1 1	0 0	0 0	0 0
1	0 0	1 0	1 0	0 1
1	0 1	1 1	1 1	1 0
1	1 0	0 1	0 1	0 0
1	1 1	0 0	0 0	0 0

## Πίνακας διεγέρσεων

Πρωτεύουσα είσοδος	Παρούσα κατάσταση	Επόμενη κατάσταση	Δευτερεύουσες έξοδοι	Πρωτεύουσες έξοδοι
<b>X</b>	<b>Q<sub>1</sub>Q<sub>2</sub></b>	<b>Q<sub>1</sub><sup>+</sup>Q<sub>2</sub><sup>+</sup></b>	<b>D<sub>1</sub>D<sub>2</sub></b>	<b>Z<sub>1</sub>Z<sub>2</sub></b>
0	0 0	0 1	0 1	0 1
0	0 1	0 0	0 0	1 0
0	1 0	0 1	0 1	0 0
0	1 1	0 0	0 0	0 0
1	0 0	1 0	1 0	0 1
1	0 1	1 1	1 1	1 0
1	1 0	0 1	0 1	0 0
1	1 1	0 0	0 0	0 0

<b>XQ<sub>1</sub></b>	00	01	11	10
<b>Q<sub>2</sub></b>	0	0	0	1
0	0	0	0	1
1	0	0	0	1

$$D_1 = X \cdot \bar{Q}_1$$

<b>XQ<sub>1</sub></b>	00	01	11	10
<b>Q<sub>2</sub></b>	0	0	0	1
0	1	1	1	0
1	0	0	0	1

$$D_2 = \bar{X} \cdot \bar{Q}_2 + Q_1 \cdot \bar{Q}_2 + X \cdot \bar{Q}_1 \cdot Q_2$$



Πίνακας διεγέρσεων

Πρωτεύουσα είσοδος	Παρούσα κατάσταση	Επόμενη κατάσταση	Δευτερεύουσες έξοδοι	Πρωτεύουσες έξοδοι
<b>X</b>	<b>Q<sub>1</sub>Q<sub>2</sub></b>	<b>Q<sub>1</sub><sup>+</sup>Q<sub>2</sub><sup>+</sup></b>	<b>D<sub>1</sub>D<sub>2</sub></b>	<b>Z<sub>1</sub>Z<sub>2</sub></b>
0	0 0	0 1	0 1	0 1
0	0 1	0 0	0 0	1 0
0	1 0	0 1	0 1	0 0
0	1 1	0 0	0 0	0 0
1	0 0	1 0	1 0	0 1
1	0 1	1 1	1 1	1 0
1	1 0	0 1	0 1	0 0
1	1 1	0 0	0 0	0 0

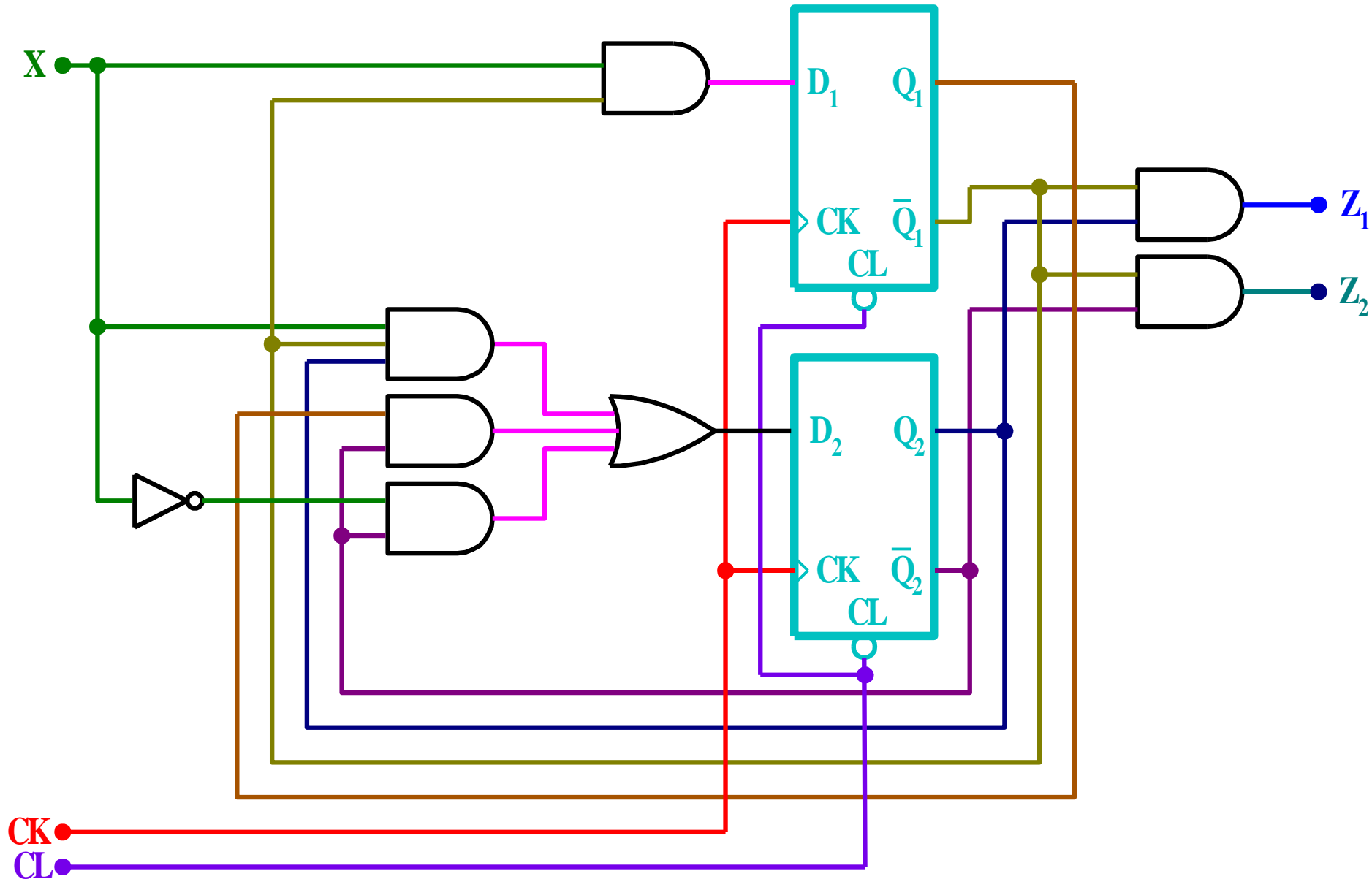
XQ <sub>1</sub>	00	01	11	10
Q <sub>2</sub>				
0	0	0	0	0
1	1	0	0	1

$$Z_1 = \bar{Q}_1 \cdot Q_2$$

XQ <sub>1</sub>	00	01	11	10
Q <sub>2</sub>				
0	1	0	0	1
1	0	0	0	0

$$Z_2 = \bar{Q}_1 \cdot \bar{Q}_2$$

$$D_1 = X \cdot \bar{Q}_1 \quad D_2 = \bar{X} \cdot \bar{Q}_2 + Q_1 \cdot \bar{Q}_2 + X \cdot \bar{Q}_1 \cdot Q_2 \quad Z_1 = \bar{Q}_1 \cdot Q_2 \quad Z_2 = \bar{Q}_1 \cdot \bar{Q}_2$$



# Παράδειγμα Traffic\_light

```
library ieee;
use ieee.std_logic_1164.all;
library altera;
USE altera.altera_primitives_components.all;

entity traffic_light is
    port(CLK, X: in std_logic;
         Z1, Z2: out std_logic);
end traffic_light;

architecture circuit of traffic_light is
    signal D1,D2,Q1,Q2, HIGH: std_logic;

begin
    D1 <= x and not Q1;
    D2 <=(not X and not Q2) or (Q1 and not Q2) or (X and not Q1 and Q2);
    FF1: DFF port map (D1,CLK, HIGH, HIGH, Q1);
    FF2: DFF port map (D2,CLK, HIGH, HIGH, Q2);
    Z1 <= (not Q1 and Q2);
    Z2 <= (not Q1 and not Q2);
end circuit;
```

# Παράδειγμα traffic\_light (2)

2. Σχεδίαση μοντέλου κυκλώματος από τον πίνακα καταστάσεων

```
LIBRARY IEEE;
```

```
USE IEEE.STD_LOGIC_1164.ALL;
```

```
ENTITY traffic_lights IS
```

```
    PORT (CK, X, RESETN    :IN STD_LOGIC;
```

```
          Z                :OUT STD_LOGIC_VECTOR (2 DOWNTO 1));
```

```
END traffic_lights;
```

```
ARCHITECTURE model OF traffic_lights IS
```

```
TYPE STATE_TYPE IS (K0, K1, K2, K3);
```

```
SIGNAL NEXTSTATE:STATE_TYPE;
```

```
    BEGIN
```

```
        PROCESS (RESETN, CK)
```

```
            BEGIN
```

```
                IF RESETN='0' THEN NEXTSTATE<=K0;
```

```
                ELSIF (CK='1' AND CK'EVENT) THEN
```

```
                    CASE NEXTSTATE IS
```

```
                        WHEN K0=>
```

```
                            IF X='0' THEN NEXTSTATE<=K1;
```

```
                            ELSE NEXTSTATE<=K2;
```

```
                            END IF; .....
```

.....

**WHEN K1=>**

**IF X='0' THEN NEXTSTATE<=K0;**

**ELSE NEXTSTATE<=K3;**

**END IF;**

**WHEN K2=>**

**IF X='0' THEN NEXTSTATE<=K1;**

**ELSE NEXTSTATE<=K1;**

**END IF;**

**WHEN K3=>**

**IF X='0' THEN NEXTSTATE<=K0;**

**ELSE NEXTSTATE<=K0;**

**END IF;**

**END CASE;**

**END IF;**

**CASE NEXTSTATE IS**

**WHEN K0 => Z<="10";**

**WHEN K1 => Z<="01";**

**WHEN K2 => Z<="00";**

**WHEN K3 => Z<="00";**

**END CASE;**

**END PROCESS;**

**END model;**

# Αποτέλεσμα προσομοίωσης.

