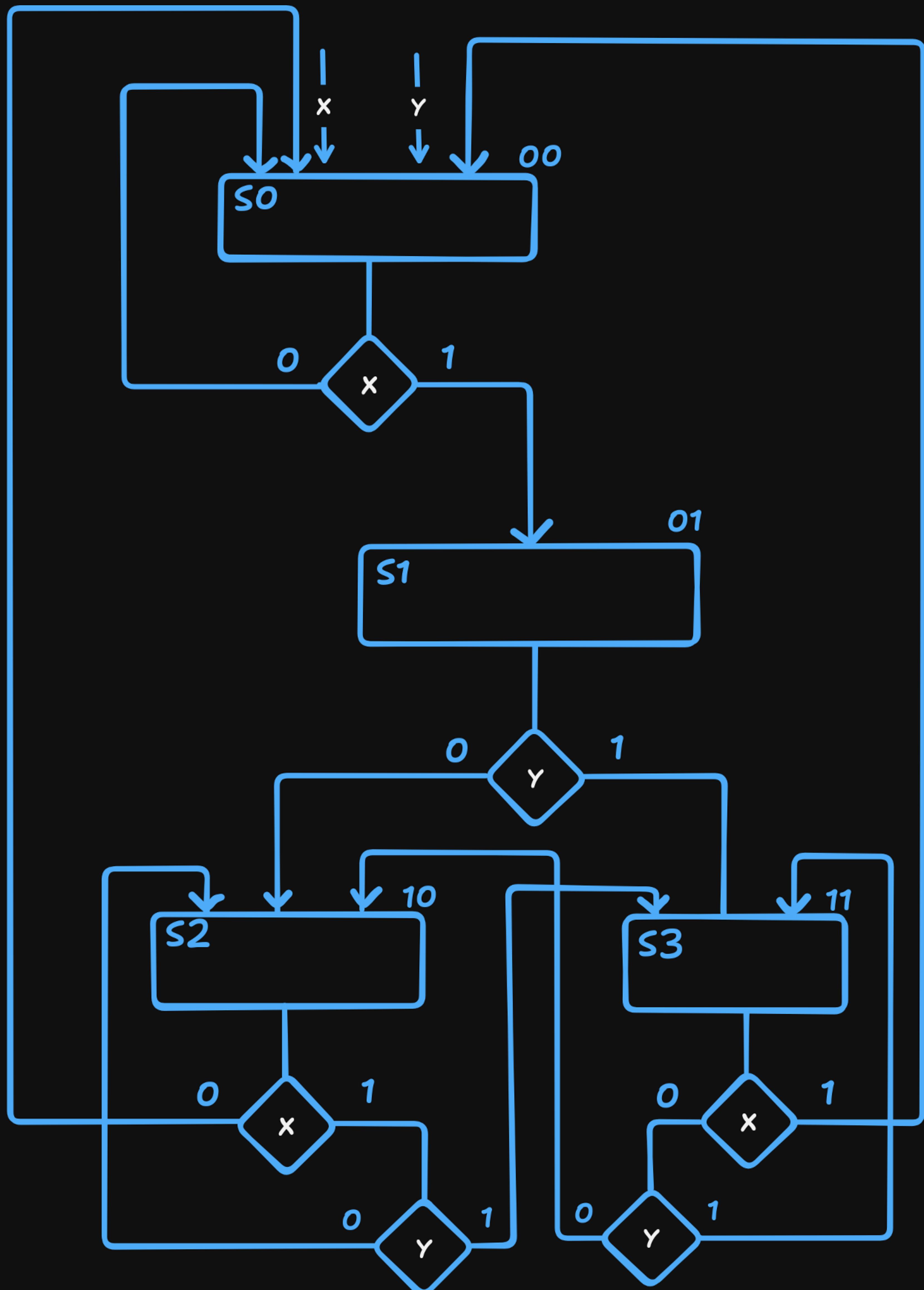


Λύσεις Ιουν. 2024 (HW2)

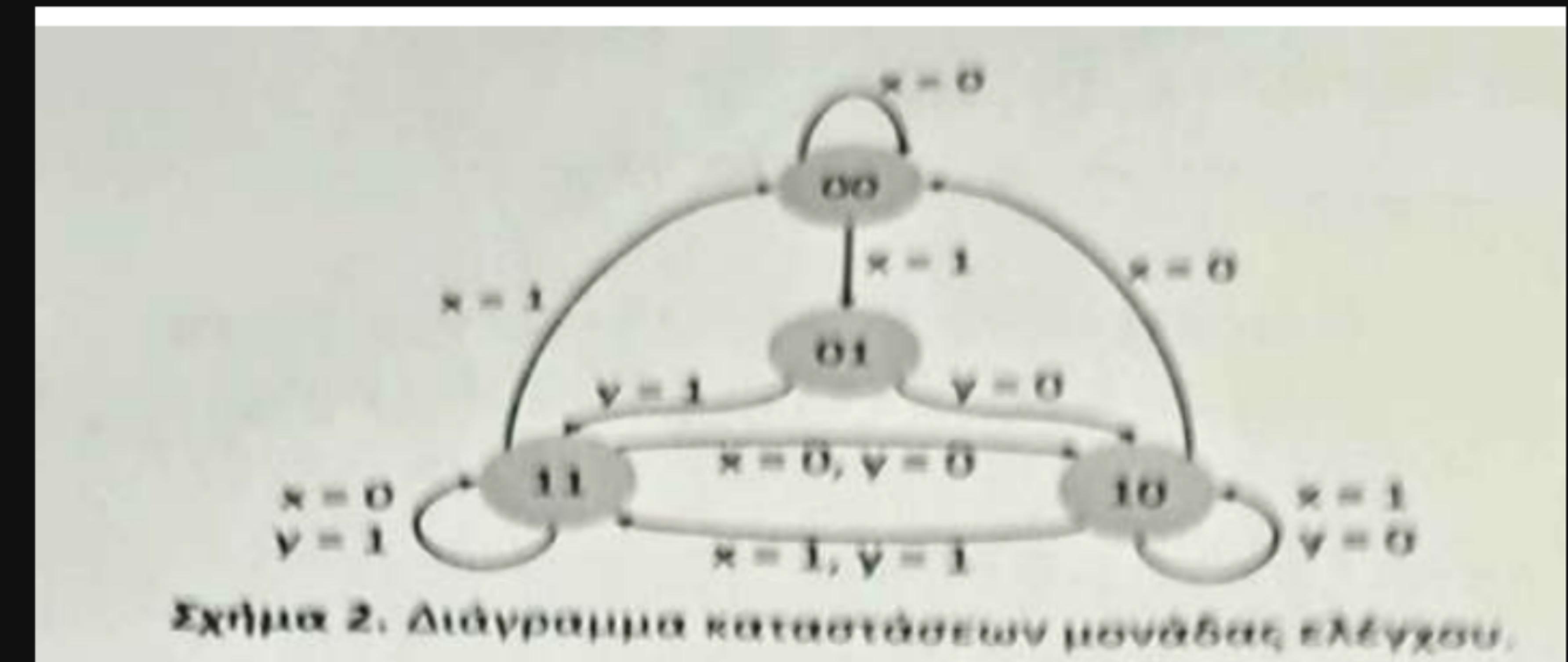


Θέμα 2: Στο Σχήμα 2, σας δίνεται ένα διάγραμμα καταστάσεων μιας μονάδας ελέγχου. Η μονάδα έχει τέσσερις καταστάσεις και δύο εισόδους x και y .

α) Παράγετε το αλγορίθμικό διάγραμμα καταστάσεων (ASM) για αυτή τη μονάδα ελέγχου. (2 μονάδες)

β) Τί τύπου διάγραμμα καταστάσεων είναι αυτό του Σχήματος 2 και πόσα FF χρειαζόμαστε για να αποθηκεύσουμε τις καταστάσεις αυτές αν χρησιμοποιήσουμε κωδικοποίηση «one-hot»; (1 μονάδα)

Χρονικότεινόμενη διάρκεια εξέτασης: 30 λεπτά



Εφόσον δεν έχουμε εξόδους, μάλλον **Moore FSM**

Χρειαζόμαστε 4 FF αφού η κάθε κατάσταση θα έχει 4 bits (*one-hot encoding*)


Λύσεις Ιουν. 2024 (HW1)

Θέμα 2: α) Σχεδιάστε έναν πολυπλέκτη 15-σε-1 χρησιμοποιώντας δύο πολυπλέκτες 8-σε-1. Διασυνδέστε τους δύο πολυπλέκτες και αναθέστε τις εισόδους $inp[0:14]$ σε αυτούς, ώστε να ελαχιστοποιείται η οποιαδήποτε επιπλέον λογική μπορεί να απαιτείται για την παραγωγή των κωδικών επιλογής. Οι επιτρεπτές τιμές των κωδικών επιλογής ($sel[3:0]$) είναι 0000 έως 1110. (2 μονάδες)

β) Σχεδιάστε εκ νέου το κύκλωμα του ερωτήματος (α), αυτή τη φορά χρησιμοποιώντας τη γλώσσα περιγραφής υλικού (HDL) Verilog. Επίσης, περιγράψτε ένα testbench το οποίο θα περιέχει και ένα κατάλληλο διάνυσμα δοκιμής για να επαληθευθεί η λογική λειτουργία του κυκλώματος. (2 μονάδες)

ΠΡΟΣΟΧΗ: Αν δεν καταφέρατε να λύσετε το ερώτημα (α) τότε μπορείτε να απαντήσετε το (β) αποκλειστικά με συμπεριφορική Verilog.

Αν έχετε λύσει το (α) (δηλαδή, έχετε παράγει το σχηματικό του πολυπλέκτη 15-σε-1) τότε μπορείτε να περιγράψετε τους πολυπλέκτες 8-σε-1 με συμπεριφορική Verilog αλλά για την επιπλέον λογική **πρέπει** να χρησιμοποιηθούν οι πρότυπες πύλες (primitives) που παρέχει η Verilog, διασυνδέοντας με δομική Verilog όλα τα υποκυκλώματα.

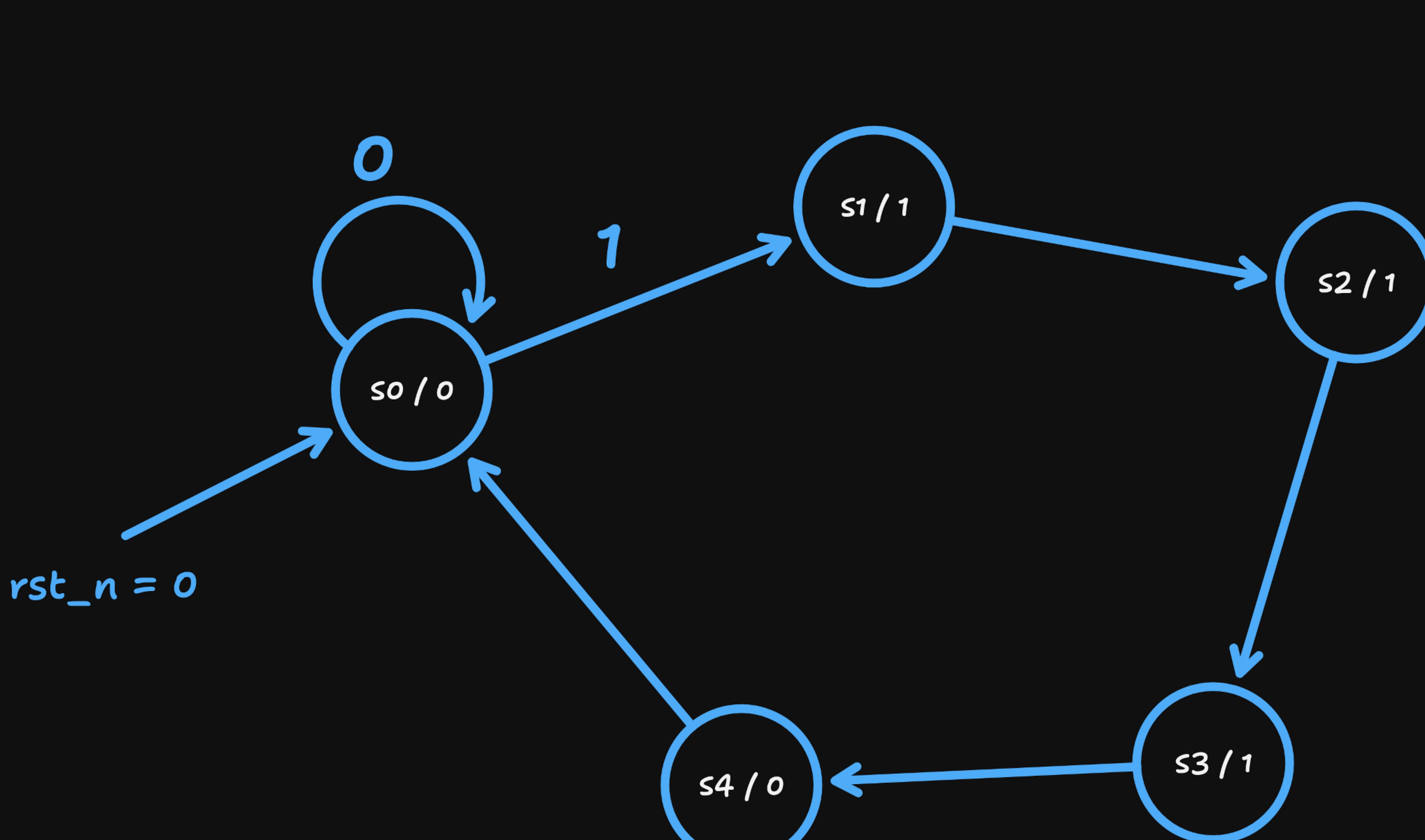


Θέμα 3: Μία μηχανή πεπερασμένων καταστάσεων με σύγχρονη λειτουργία έχει μία είσοδο, X και μία έξοδο, Y . Όταν η είσοδος X αλλάζει από 0 σε 1, η έξοδος Y πρέπει να ενεργοποιείται για τρεις κύκλους, ανεξάρτητα από την τιμή της εισόδου, και κατόπιν να απενεργοποιείται για δύο κύκλους πριν η μηχανή αποκριθεί σε μία επόμενη ενεργοποίηση της X . Η μηχανή πρέπει να έχει τη δυνατότητα «ενεργού-χαμηλά» σύγχρονου μηδενισμού (rst_n).

α) Να σχεδιαστεί το διάγραμμα καταστάσεων της μηχανής. (1 μονάδα)

β) Περιγράψτε τη μηχανή αυτή με τη γλώσσα περιγραφής υλικού (HDL) Verilog σύμφωνα με τη μεθοδολογία που διδαχθήκατε στα πλαίσια του μαθήματος. (2 μονάδες)

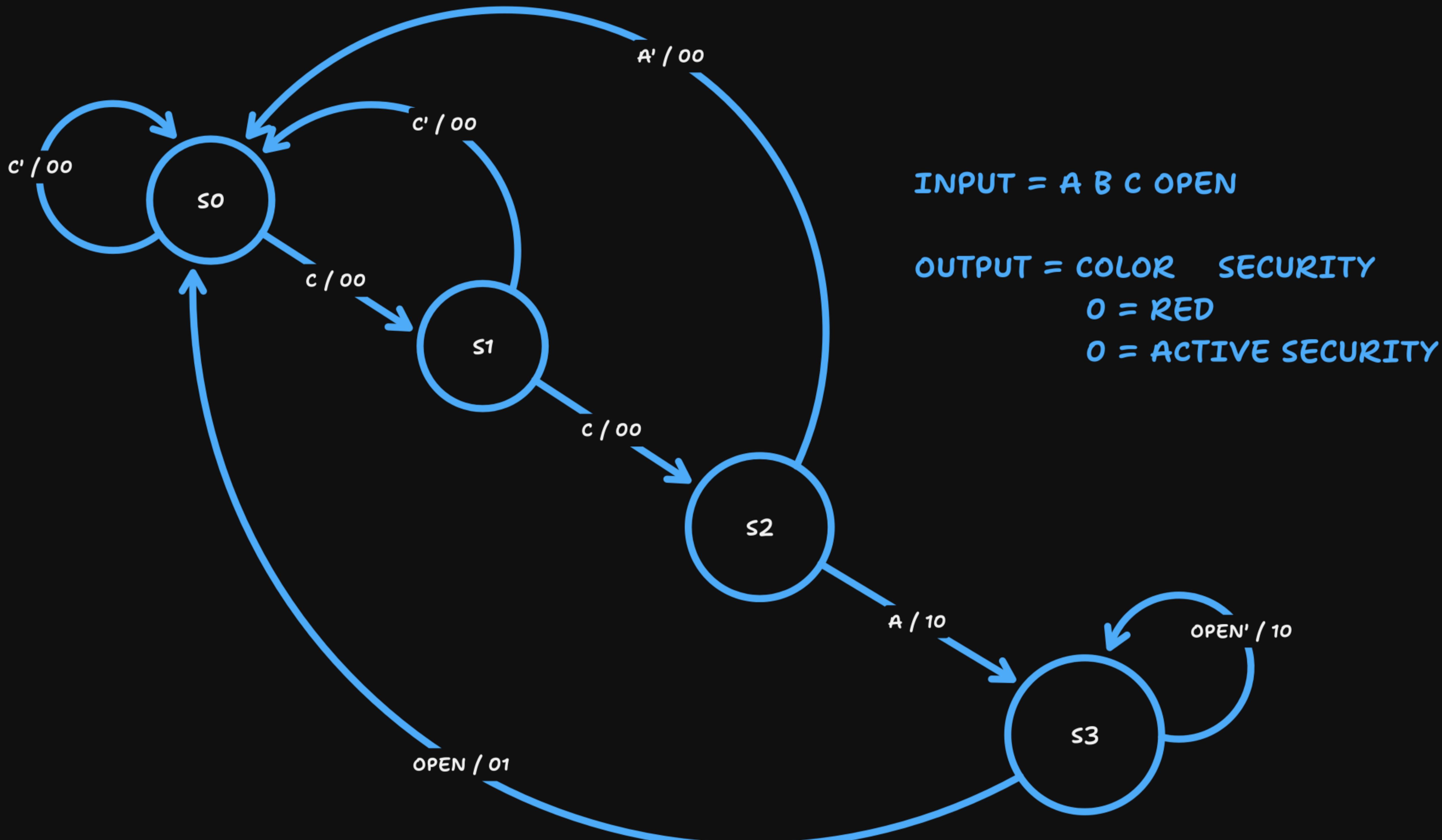
γ) Παρέχετε ένα testbench το οποίο θα περιέχει ένα κατάλληλο διάνυσμα λογικής για την επαλήθευση της μηχανής. (1 μονάδα)



Θέμα 3: Μία ηλεκτρονική κλειδαριά διαθέτει τέσσερα κουμπιά (A, B, C, OPEN) και απαιτεί μία συγκεκριμένη αλληλουχία τριών (3) σωστών γραμμάτων (A, B, ή C) για να ξεκλειδώσει. Στην περίπτωση λανθασμένου γράμματος, η κλειδαριά επανέρχεται στην αρχική της κατάσταση περιμένοντας τη σωστή αλληλουχία ψηφίων. Καθ' όλη τη διάρκεια ο λαμπτήρας της κλειδαριάς έχει κόκκινη ένδειξη, ενώ όταν εισαχθεί ο σωστός κωδικός γίνεται πράσινη. Μόλις η ένδειξη γίνει πράσινη και πατηθεί το κουμπί «OPEN», η ασφάλεια απενεργοποιείται (η κλειδαριά ξεκλειδώνει) και η ένδειξη γίνεται κόκκινη. Έπειτα, η κλειδαριά κλειδώνει αυτόματα (χωρίς κάποια είσοδο δηλαδή από το χρήστη), επανέρχεται στην αρχική της κατάσταση, και αναμένει εκ νέου τη σωστή αλληλουχία. Ο κωδικός για να ξεκλειδώσει η κλειδαριά είναι «CCA».

Σχεδιάστε το διάγραμμα της μηχανής πεπερασμένων καταστάσεων (FSM) και διατυπώστε τις λογικές συναρτήσεις των επόμενων καταστάσεων και των εξόδων (ένδειξη και ασφάλεια) της ηλεκτρονικής κλειδαριάς.

(2,5 μονάδες)



Έστω:

- $s_0 = 00$
- $s_1 = 01$
- $s_2 = 10$
- $s_3 = 11$

Present state		Input				Next state	
D_1	D_0	A	B	C	OPEN	D_1'	D_0'
0	0	X	X	1	X	0	1
0	0	X	X	0	X	0	0
0	1	X	X	1	X	1	0
0	1	X	X	0	X	0	0
1	0	1	X	X	X	1	1
1	0	0	X	X	X	0	0
1	1	X	X	X	1	0	0
1	1	X	X	X	0	1	1

$$D_1' = \bar{D}_1 D_0 C + D_1 \bar{D}_0 A + D_1 D_0 \bar{OPEN}$$

$$D_0' = \bar{D}_1 \bar{D}_0 C + D_1 \bar{D}_0 A + D_1 D_0 \bar{OPEN}$$

Present state		Input				Output	
D_1	D_0	A	B	C	OPEN	COLOR	SEC
0	0	X	X	1	X	0	0
0	0	X	X	0	X	0	0
0	1	X	X	1	X	0	0
0	1	X	X	0	X	0	0
1	0	1	X	X	X	1	0
1	0	0	X	X	X	0	0
1	1	X	X	X	1	0	1
1	1	X	X	X	0	1	0

$$\text{COLOR} = D_1 \bar{D}_0 A + D_1 D_0 \bar{OPEN}$$

$$\text{SECURITY} = D_1 D_0 \bar{OPEN}$$

Λύσεις Ιουν. 2022 (HW2)

$$F = \underbrace{A'BC'}_{\text{red}} + \underbrace{BD}_{\text{orange}} + \underbrace{ACD'}_{\text{green}} + \underbrace{B'C'D'}_{\text{blue}}$$

συγχρένη μεταβλητή B

Επίκαια Λογικής II

Δευτέρα, 27 Ιουνίου 2022

Θέμα 1: Η παρακάτω έκφραση περιέχει κάποια στατικά hazards.

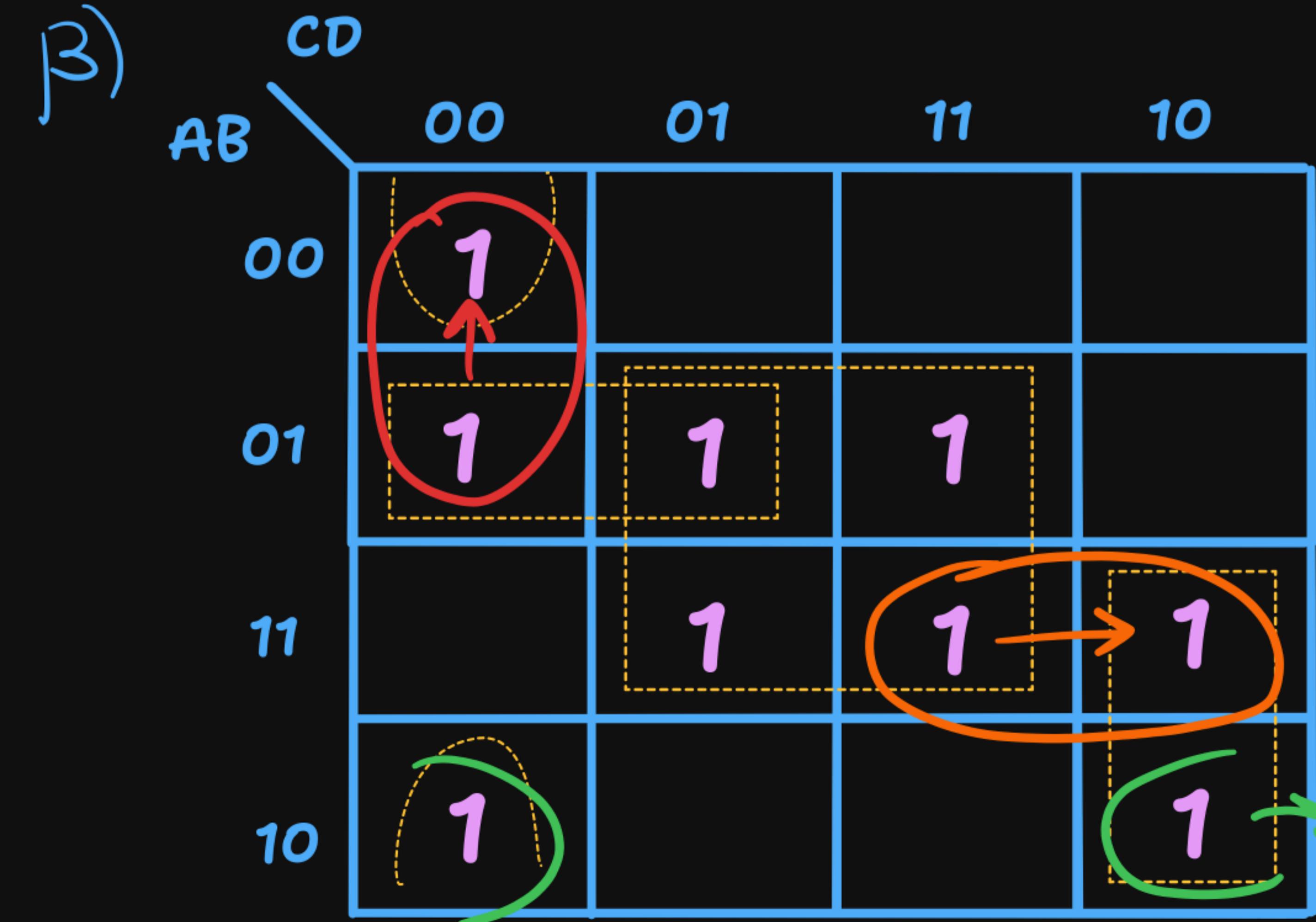
$$F = \bar{A}B\bar{C} + BD + A\bar{C}\bar{D} + \bar{B}\bar{C}\bar{D}, \quad (1)$$

- (α) Προσδιορίστε ποιά είναι τα hazards αυτά (με απλή επόπτευση). (0.5 βαθμός)

- (β) Δείξτε τους RPI που προκαλούν τα hazards αυτά σε έναν κατάλληλο χάρτη Karnaugh. (0.5 βαθμός)
- (γ) Γράψτε ποιές μεταβάσεις μπορεί να οδηγήσουν σε αυτά τα hazards. (0.5 βαθμός)
- (δ) Δώστε τη νέα συνάρτηση F που εμπεριέχει προστασία από τα hazards αυτά. (0.5 βαθμός)

Διάρκεια εξέτασης: 30 λεπτά

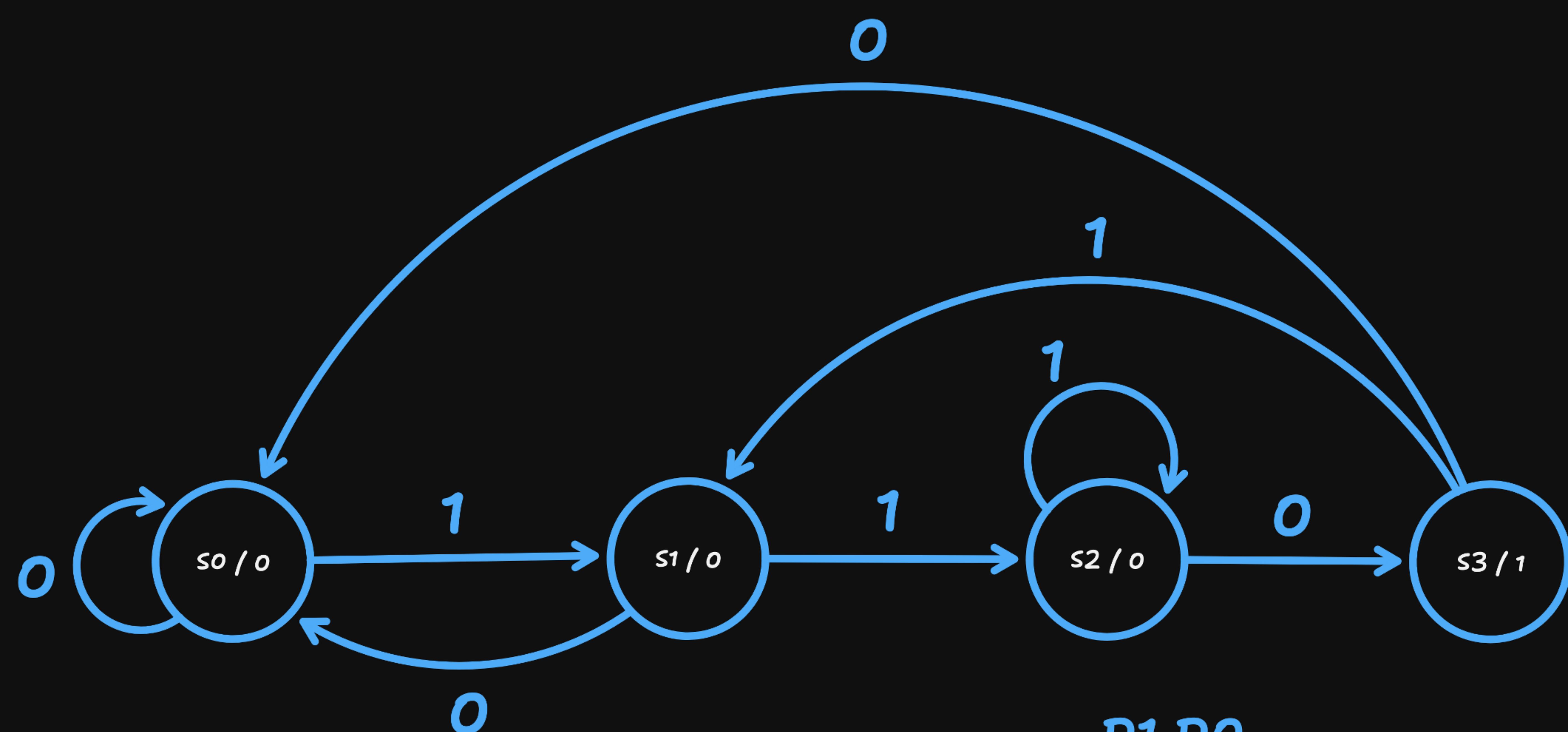
a) Συγχρένες μεταβλητές είναι οι B, C, D



- γ) $1111 \rightarrow 1110 \quad (\text{D})$
- $1010 \rightarrow 1000 \quad (\text{C})$
- $0100 \rightarrow 0000 \quad (\text{B})$
- δ) $F = A'BC' + BD + ACD' + B'C'D' + ABC + A'C'D'$
 $+ AB'D'$

Λύσεις Φεβρ. 2022 (HW1)

Θέμα 2: Σχεδιάστε μια σύγχρονη μηχανή πεπερασμένων καταστάσεων (FSM) τύπου Moore η οποία ανιχνεύει την είσοδο 110 (όταν αυτό συμβαίνει η έξοδος θα γίνει «1», και για ένα μόνο κύκλο).
 α. Δείξτε το διάγραμμα καταστάσεων και το πίνακα καταστάσεων (1.5 βαθμοί).
 β. Σχεδιάστε το λογικό διάγραμμα της FSM με D Flip Flops και πύλες (ότι είδους χρειάζεται). Δείξτε καθαρά τους πίνακες Karnaugh και τις εξισώσεις για τις εισόδους D (1.5 βαθμοί).



CURRENT STATE	INPUT		NEXT STATE
D1	DO	D1'	DO'
0	0	0	0 0
0	0	1	0 1
0	1	0	0 0
0	1	1	1 0
1	0	0	1 1
1	0	1	1 0
1	1	0	0 0
1	1	1	0 1

D1'	DO'	00	01	11	10
0	0				1
1	1		1		1

$$D1' = D1 \cdot \overline{DO} + X \cdot \overline{D1} \cdot DO$$

D1'	DO'	00	01	11	10
0	0				1
1	1	1			1

$$DO' = X \cdot \overline{D1} \cdot \overline{DO} + X \cdot D1 \cdot DO + \overline{X} \cdot D1 \cdot \overline{DO}$$

OUTPUT		
D1	DO	Y
0	0	0
0	1	0
1	0	0
1	1	1

$$Y = D1 \cdot DO$$

