

Συστήματα Μικροϋπολογιστών 2018-2019

2η Ομάδα Ασκήσεων

1η Άσκηση

```
IN 10H
MVI A, 00H ;Αριθμοί (0<=A<=255)
LXI H, 0900H ;Διευθύνσεις
LXI B, 0000H ;Μονάδες
MVI D, 00H ;Αριθμοί [10H, 60H]

START:
MOV M, A ;Αποθήκευση στη μνήμη
MOV E, A ;Προσωρινή αποθήκευση
JMP ONES

COUNTONES:
MOV A, E
JMP NUMBERS

COUNTNUM:
INX H ;Επόμενη θέση μνήμης
INR A ;Επόμενος αριθμός
CPI 00H ;A=00H ?
JZ FINISH ;Έλεγχος τερματισμού
JMP START

ONES: ;Μέτρηση μονάδων (δυαδικών 1)
STC ;CY=0
CMC ;
RAR ;Ψηφίο-ψηφίο στο CY
JNC ONE ;Μονάδα?
INX B

ONE:
CPI 00H ;A=00H ?
JZ COUNTONES;Έλεγχος τερματισμού
JMP ONES

NUMBERS: ;Μέτρηση αριθμών στο [10H, 60H]
CPI 10H
JC COUNTNUM ;A<10H ?
CPI 60H
JC NUMBER ;A<60H ?
JNZ COUNTNUM;A=60H ?

NUMBER:
INR D
JMP COUNTNUM

FINISH:
END
```

2η Άσκηση

```

    MVI A, FFH
    STA 3000H
    MVI D, 96H          ;Χρονόμετρο 30 secs=150*0,2 secs (150=96H)
    LXI B, 00C8H        ;Καθυστερήση 200 ms
START:
    LDA 2000H           ;Διάβασμα των διακοπών
    ANI 80H             ;MSB
    CPI 00H             ;OFF ?
    JZ OFF1
    JMP START
OFF1:
    ;1ο OFF
    LDA 2000H
    ANI 80H
    CPI 80H            ;ON ?
    JZ ON1
    JMP OFF1
ON1:
    ;1ο ON
    LDA 2000H
    ANI 80H
    CPI 00H            ;OFF ?
    JZ OFF2
    JMP ON1
OFF2:
    ;2ο OFF
    LDA 2000H
    ANI 80H
    CPI 80H            ;ON ?
    JZ ON2
    MVI A, 00H          ;Αναμμα
    STA 3000H           ;
    CALL DELB           ;Χρόνος
    DCR D               ;
    MOV A, D
    CPI 00H             ;Εκπνοή χρόνου ?
    JNZ OFF2            ;Έλεγχος εκπνοής χρόνου
    MVI A, FFH          ;Σβήσιμο
    STA 3000H           ;
    MVI D, 96H          ;Επαν-αρχικοποίηση του χρονομέτρου
    JMP OFF1
ON2:
    ;2ο ON
    LDA 2000H
    ANI 80H
    CPI 00H            ;OFF ?
    JZ RESTART          ;Έλεγχος ανανέωσης χρόνου
    MVI A, 00H
    STA 3000H
    CALL DELB
    DCR D
    MOV A, D
    CPI 00H
    JNZ ON2
    MVI A, FFH
    STA 3000H
    MVI D, 96H
    JMP OFF1
RESTART:
    MVI D, 96H          ;Επαν-αρχικοποίηση του χρονομέτρου
    JMP OFF2

END
```

3η Άσκηση

(i)

```
START:
    LDA 2000H    ;Διακόπτες
    MVI B, FFH   ;LEDs
SCAN:
    RRC         ;Σάρωση για μονάδες
    MOV C, A     ;CY <- LSB
    JC FOUND     ;Προσωρινή αποθήκευση
    CPI 00H      ;Έλεγχος για μονάδα
    JZ ZEROS     ;A=00000000 ?
    MOV A, B     ;Έλεγχος για απουσία μονάδων στην είσοδο
    STC         ;CY <- 0
    CMC         ;
    RAL         ;Σβήσιμο του αντίστοιχου LED
    MOV B, A     ;Προσωρινή αποθήκευση
    MOV A, C
    JMP SCAN
FOUND:
    MOV A, B     ;Βρέθηκε η 1η μονάδα
    CMA         ;Αναμνα LEDs ίσης και υψηλότερης τάξης
    STA 3000H    ;Αντίστροφη λογική
    JMP START
ZEROS:
    MVI A, FFH   ;Μόνο μηδενικά
    STA 3000H    ;Σβήσιμο LEDs
    JMP START
END
```

(ii)

```
START:
    MVI B, 80H   ;LED
    CALL KIND    ;Διάβασμα πληκτρολογίου
    CPI 00H      ;A=0 ?
    JZ TURNOFF   ;Έλεγχος για το 0
    CPI 09H      ;A>=9 ?
    JNC TURNOFF  ;Έλεγχος για πλήκτρο >=9
SLIDE:
    DCR A        ;Μείωση μέχρι να μηδενιστεί
    MOV C, A
    MOV A, B
    RLC         ;Ολίσθηση μέχρι το επιθυμητό LED
    MOV B, A
    MOV A, C
    JZ SHOW     ;Έλεγχος για μηδενισμό
    JMP SLIDE
TURNOFF:
    MVI B, 00H   ;Σβήσιμο LED
    MOV A, B
    CMA         ;Πλήκτρο >=9 ή 0
    STA 3000H    ;Αναμνα LED
    JMP START
SHOW:
    MOV A, B
    CMA         ;Αντίστροφη λογική
    STA 3000H
    JMP START
END
```

(iii) Δίνεται παρακάτω ένα ενδεικτικό πρόγραμμα μόνο για τις γραμμές 1, 6 και 7 του πληκτρολογίου. Οι υπόλοιπες γραμμές προγραμματίζονται όμοια. Η υλοποίηση έγινε με εκτέλεση εντολών σειριακά και όχι αλγοριθμικά, λόγω του περιορισμένου πλήθους καταχωρητών. Έγινε προσπάθεια να μειωθούν οι γραμμές κώδικα στο ελάχιστο αλλά και ταυτόχρονα να εμφανίζονται σωστά τα μηνύματα στην οθόνη του προσομοιωτή, γι' αυτό κάποιες εντολές επαναλαμβάνονται πολλές φορές. Το πρόγραμμα εμφανίζει όλα τα πλήκτρα που υλοποιεί εκτός του A.

```

IN 10H
LXI H, 08FAH      ;Τα μηνύματα στις διευθύνσεις 08FAH-08FFH
MVI M, 10H        ;Αρχικό σβήσιμο της οθόνης
LXI H, 08FBH      ;
MVI M, 10H        ;
LXI H, 08FCH      ;
MVI M, 10H        ;
LXI H, 08FDH      ;
MVI M, 10H        ;
LXI H, 08FEH      ;
MVI M, 10H        ;
LXI H, 08FFH      ;
MVI M, 10H        ;

SCAN7:
MVI A, 7FH        ;Γραμμή 7 (πλήκτρα D, E, F)
STA 2800H         ;Σάρωση
LDA 1800H         ;Ανάγνωση
MVI B, 07H
ANA B

CPI 03H           ;Στήλη 2
JZ KEYF          ;Έλεγχος για το πλήκτρο F
CPI 05H           ;Στήλη 1
JZ KEYE          ;Έλεγχος για το πλήκτρο E
CPI 06H           ;Στήλη 0
JZ KEYD          ;Έλεγχος για το πλήκτρο D

CALL SHOW
JMP SCAN1

KEYF:             ;Εμφάνιση πλήκτρου F
LXI H, 08FFH
MVI M, 00H        ;0
LXI H, 08FEH
MVI M, 0FH        ;F
CALL SHOW
JMP SCAN1

KEYE:             ;Εμφάνιση πλήκτρου E
LXI H, 08FFH
MVI M, 00H        ;0
LXI H, 08FEH
MVI M, 0EH        ;E
CALL SHOW
JMP SCAN1

KEYD:             ;Εμφάνιση πλήκτρου D
LXI H, 08FFH
MVI M, 00H        ;0
LXI H, 08FEH
MVI M, 0DH        ;D
CALL SHOW

SCAN1:
MVI A, FDH        ;Γραμμή 1 (πλήκτρα FETCH ADRS, FETCH REG, RUN)
STA 2800H
LDA 1800H
MVI B, 07H
ANA B

CPI 03H           ;Έλεγχος για το πλήκτρο FETCH ADRS
JZ KEYFETCHADRS  ;Έλεγχος για το πλήκτρο FETCH REG
CPI 05H           ;Έλεγχος για το πλήκτρο FETCH REG
JZ KEYFETCHREG   ;Έλεγχος για το πλήκτρο RUN
CPI 06H           ;Έλεγχος για το πλήκτρο RUN
JZ KEYRUN

CALL SHOW
JMP SCAN6

KEYFETCHADRS:     ;Εμφάνιση πλήκτρου FETCH ADRS

```

```

    LXI H,08FFH
    MVI M,08H           ;8
    LXI H,08FEH
    MVI M,02H           ;2
    CALL SHOW
    JMP SCAN6
KEYFETCHREG:           ;Εμφάνιση πλήκτρου FETCH REG
    LXI H,08FFH
    MVI M,08H           ;8
    LXI H,08FEH
    MVI M,00H           ;0
    CALL SHOW
    JMP SCAN6
KEYRUN:                ;Εμφάνιση πλήκτρου RUN
    LXI H,08FFH
    MVI M,08H           ;8
    LXI H,08FEH
    MVI M,04H           ;4
    CALL SHOW

SCAN6:
    MVI A,BFH           ;Γραμμή 6 (πλήκτρα A, B, C)
    STA 2800H
    LDA 1800H
    MVI B,07H
    ANA B

    CPI 03H
    JZ KEYC             ;Έλεγχος για το πλήκτρο C
    CPI 05H
    JZ KEYB             ;Έλεγχος για το πλήκτρο B
    CPI 06H
    JZ KEYA             ;Έλεγχος για το πλήκτρο A

    CALL SHOW
    JMP SCAN7
KEYC:                  ;Εμφάνιση πλήκτρου C
    LXI H,08FFH
    MVI M,00H           ;0
    LXI H,08FEH
    MVI M,0CH           ;C
    CALL SHOW
    JMP SCAN7
KEYB:                  ;Εμφάνιση πλήκτρου B
    LXI H,08FFH
    MVI M,00H           ;0
    LXI H,08FEH
    MVI M,0BH           ;B
    CALL SHOW
    JMP SCAN7
KEYA:                  ;Εμφάνιση πλήκτρου A
    LXI H,08FFH
    MVI M,00H           ;0
    LXI H,08FEH
    MVI M,0AH           ;A
    CALL SHOW
    JMP SCAN7

SHOW:                  ;Εμφάνιση στην οθόνη
    LXI D,08FAH
    CALL STDM
    CALL DCD
    RET

END

```

4η Άσκηση

START:

```
MVI D, 00H      ; LEDs
                 ; X3
LDA 2000H        ; A3
ANI 80H          ; 10000000
RRC              ; Ολίσθηση για λογική πράξη με το επόμενο ψηφίο
MOV B, A         ; Προσωρινή αποθήκευση
LDA 2000H        ; B3
ANI 40H          ; 01000000
ANA B           ; A3 AND B3
RRC              ; Ολίσθηση για αποθήκευση στο X3
RRC              ;
RRC              ;
MOV D, A         ; Αποθήκευση
                 ; X2
LDA 2000H        ; A2
ANI 20H          ; 00100000
RRC              ;
MOV B, A         ;
LDA 2000H        ; B2
ANI 10H          ; 00010000
ANA B           ; A2 AND B2
RRC              ; Ολίσθηση για αποθήκευση στο X2
RRC              ;
ORA D            ; Προσθήκη
MOV D, A         ;
                 ; X1
LDA 2000H        ; A1
ANI 08H          ; 00001000
RRC              ;
MOV B, A         ;
LDA 2000H        ; B1
ANI 04H          ; 00000100
ORA B           ; A1 OR B1
RRC              ; Ολίσθηση στο LSB για XOR
RRC              ;
MOV C, A         ; Προσωρινή αποθήκευση
RLC              ; Επαναφορά για αποθήκευση στο X1
ORA D            ;
MOV D, A         ;
                 ; X0
LDA 2000H        ; A0
ANI 02H          ; 00000010
RRC              ;
MOV B, A         ;
LDA 2000H        ; B0
ANI 01H          ; 00000001
ORA B           ; A0 OR B0
XRA C            ; (A0 OR B0) XOR X1
ORA D            ;

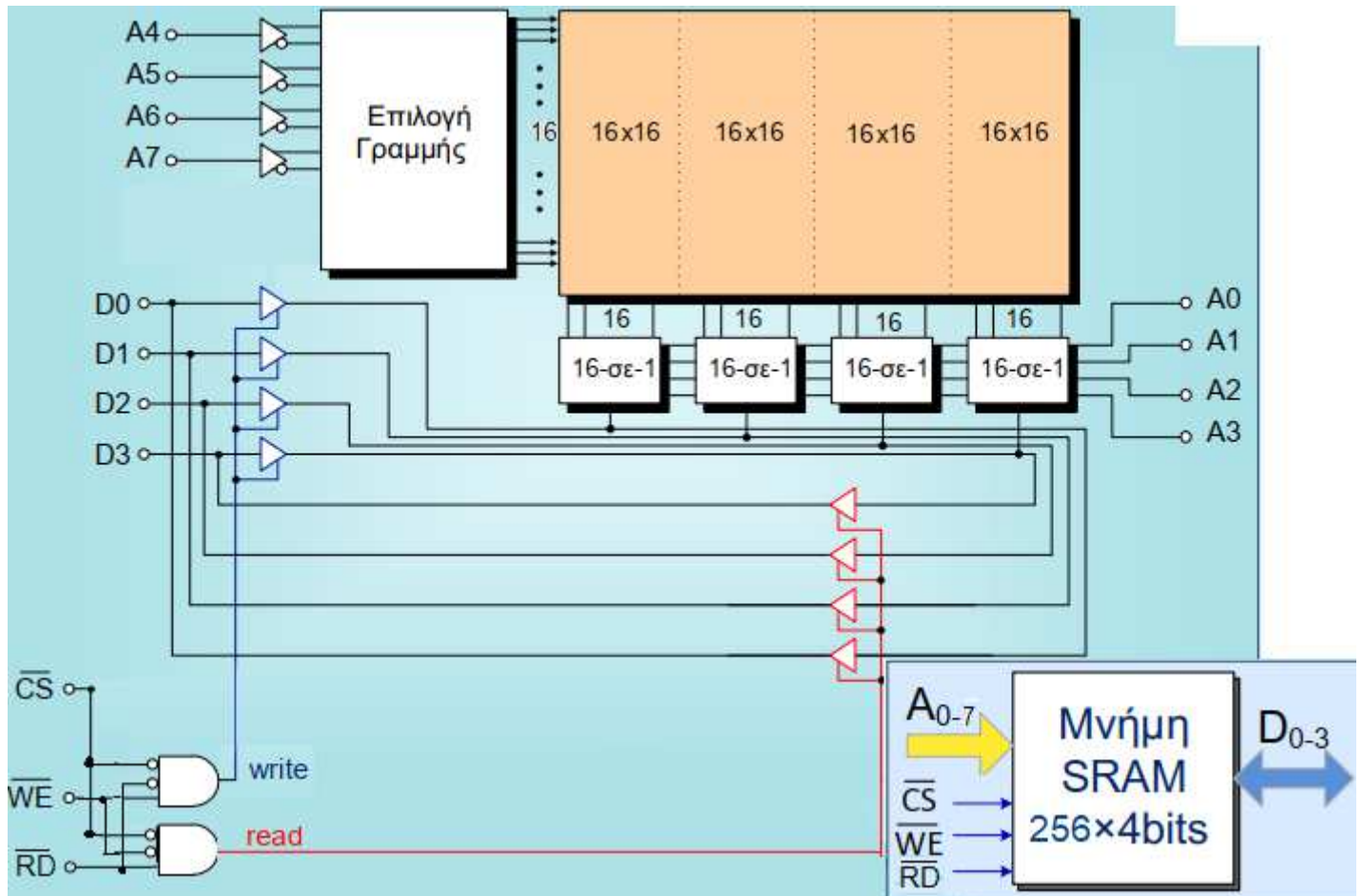
CMA              ; Αντίστροφη λογική
STA 3000H
JMP START
```

END

5η Άσκηση

Δίνεται ένα ενδεικτικό σχήμα της δομής μιας μνήμης SRAM 256x4 bits μετά από τροποποίηση του σχήματος 3.2 του βιβλίου θεωρίας ([Συστήματα Μικροϋπολογιστών](#), Κ. Πεκμεσιζή, Εκδόσεις Συμμετρία, 1995). Οι διαφορές με το βιβλίο είναι τα λιγότερα bits για τις γραμμές διευθύνσεων και τη γραμμή του πίνακα μετά το κύκλωμα αποκωδικοποίησης και το σήμα RD για την ανάγνωση.

Η λειτουργία της μνήμης πρέπει να ενεργοποιηθεί με το \overline{CS} . Έπειτα, επιλέγουμε να διαβάσουμε τη μνήμη με το \overline{RD} , που μέσω των κόκκινων πυλών οδηγεί τα δεδομένα από τους πολυπλέκτες στην έξοδο (D_0-D_3) ή να γράψουμε σε αυτήν με το \overline{WE} , που μέσω των μπλε πυλών οδηγεί τα 4 bits της εισόδου (D_0-D_3) στους πολυπλέκτες για να επιλέξουν την επιθυμητή τετράδα του πίνακα. Η επιλογή από τις 16 τετράδες γίνεται με τα πρώτα 4 bits της διεύθυνσης (A_0-A_3), ενώ η γραμμή του πίνακα στην οποία ανήκει η τετράδα επιλέγεται με τα τελευταία 4 bits της διεύθυνσης (A_4-A_7).

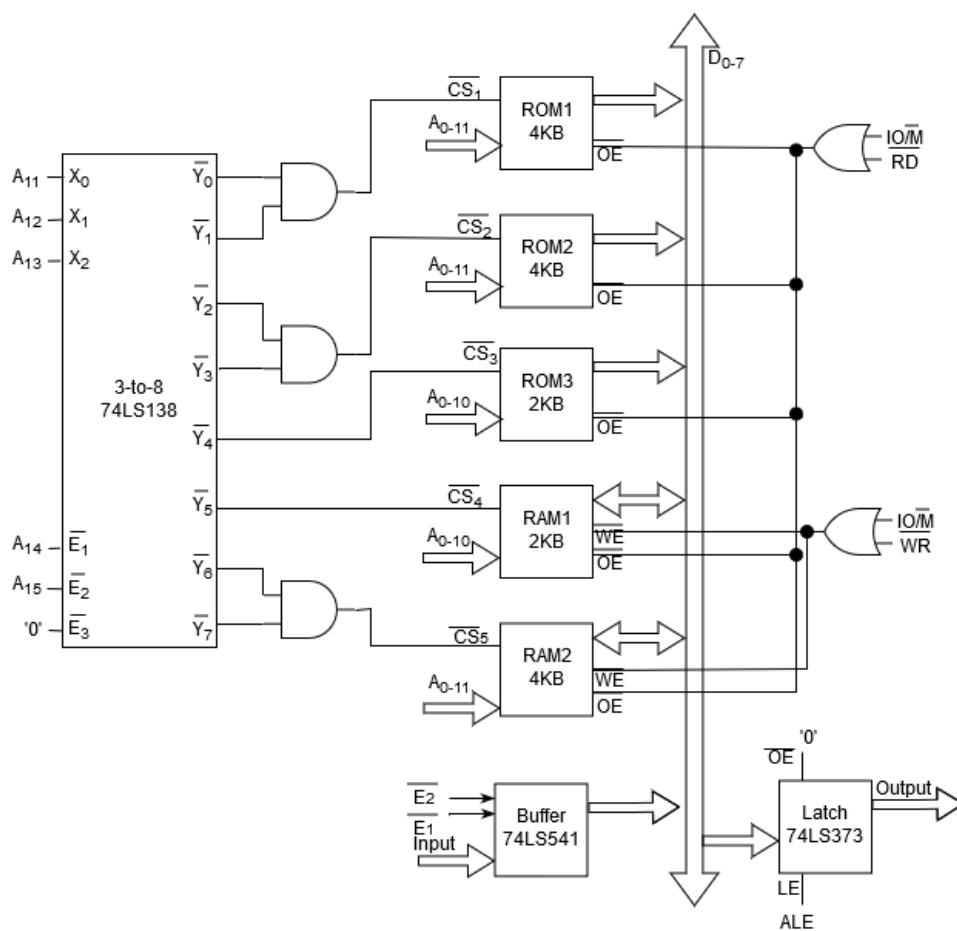


6η Άσκηση

Χάρτης μνήμης

Μνήμη	Διεύθυνση	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ROM1 - 4K	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0FFF	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
ROM2 - 4K	1000	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	1FFF	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
ROM3 - 2K	2000	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	27FF	0	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1
RAM1 - 2K	2800	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
	2FFF	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
RAM2 - 4K	3000	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
	3FFF	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Λογικό διάγραμμα

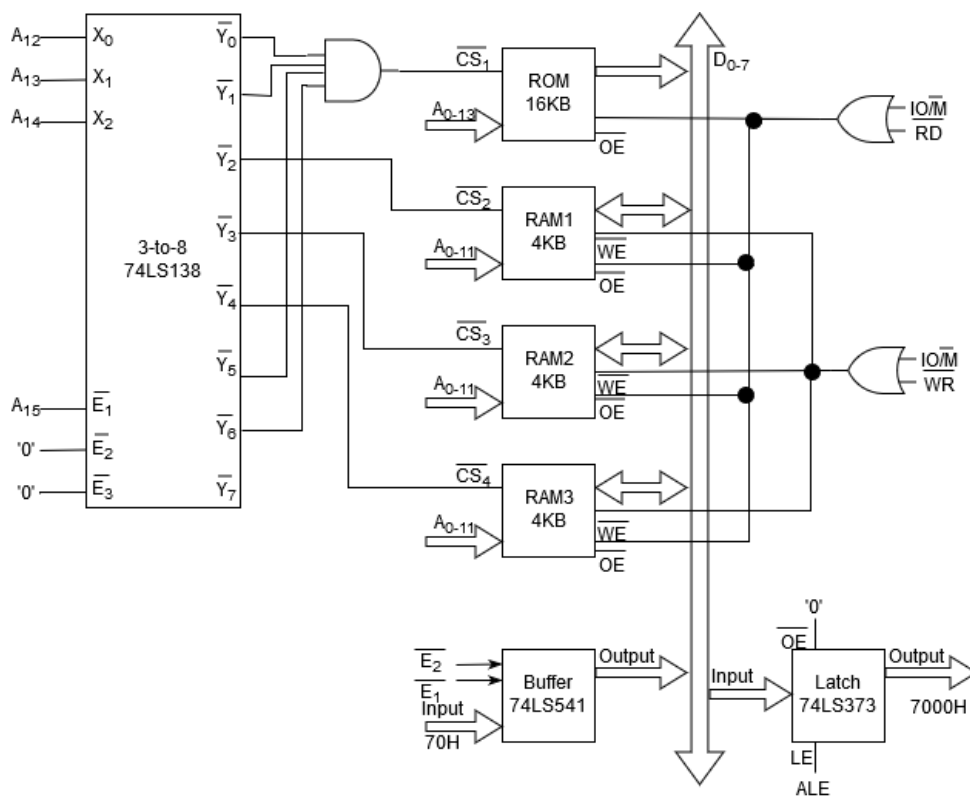


7η Άσκηση

Χάρτης μνήμης

Μνήμη	Διεύθυνση	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ROM - 8K	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1FFF	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
RAM1 - 4K	2000	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	2FFF	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
RAM2 - 4K	3000	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
	3FFF	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
RAM3 - 4K	4000	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4FFF	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1
ROM - 8K	5000	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	6FFF	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1

Λογικό διάγραμμα



Σημείωση: Η επεξεργασία του κώδικα των προγραμμάτων έγινε στο [Notepad++](#) 7.6.6 portable, η επεξεργασία του σχήματος της 5ης Άσκησης έγινε στο [Paint](#) των Windows 10, η επεξεργασία του σχήματος της 6ης και 7ης Άσκησης έγινε στο [draw.io](#) 11.3.0 και η συγγραφή της παρούσας αναφοράς έγινε στο [LibreOffice](#) 6.3.1 portable.