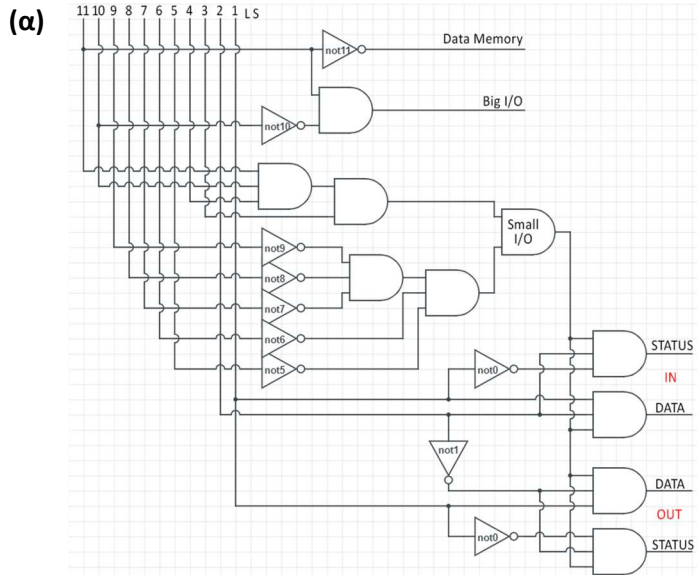


## Ασκηση 13

ΗΥ225-Οργάνωση Υπολογιστών

csd4569-Χρηστος Παπασταμος

### Ασκηση 13.4



(β)

HEX	BIN	Θεση
000	000 0000 0000	Λεξη μνημης 0
00E	000 0000 1110	Λεξη μνημης 14
100	001 0000 0000	Λεξη μνημης 256
200	010 0000 0000	Λεξη μνημης 512
2FF	010 1111 1111	Λεξη μνημης 767
300	011 0000 0000	Λεξη μνημης 768
3FF	011 1111 1111	Λεξη μνημης 1023
400	100 0000 0000	Θεση bufer 0
4FF	100 1111 1111	Θεση bufer 255
500	101 0000 0000	Θεση bufer 256
5FF	101 1111 1111	Θεση bufer 511
600	110 0000 0000	Παρανομη θεση
608	110 0000 1000	Παρανομη θεση
60A	110 0000 1010	Παρανομη θεση
60E	110 0000 1110	State (OUT)
60F	110 0000 1111	Data (OUT)

HEX	BIN	Θεση
610	110 0001 0000	Παρανομη θεση
680	110 1000 0000	Παρανομη θεση
6E0	110 1110 0000	Παρανομη θεση
6E3	110 1110 0010	Παρανομη θεση
6EF	110 1110 1111	Παρανομη θεση
6FF	110 1111 1111	Παρανομη θεση
700	111 0000 0000	Παρανομη θεση
708	111 0000 1000	Παρανομη θεση
70A	111 0000 1010	Παρανομη θεση
70E	111 0000 1110	Παρανομη θεση
70F	111 0000 1111	Παρανομη θεση
710	111 0001 0000	Παρανομη θεση
780	111 1000 0000	Παρανομη θεση
7C0	111 1100 0000	Παρανομη θεση
7C3	111 1100 0011	Παρανομη θεση
7CF	111 1100 1111	Παρανομη θεση
7FF	111 1111 1111	Παρανομη θεση
800	000 0000 0000	Λεξη μνημης 0
808	000 0000 1000	Λεξη μνημης 8
80A	000 0000 1010	Λεξη μνημης 10
80E	000 0000 1110	Λεξη μνημης 14
80F	000 0000 1111	Λεξη μνημης 15
C0E	100 0000 1110	Θεση bufer 14
C0F	100 0000 1111	Θεση bufer 15
F02	111 0000 0010	Παρανομη θεση
F12	111 0001 0010	Παρανομη θεση

### **Ασκηση 13.5**

A) char read\_kbd\_busywait\_char(){

```
    char *data;

    int *state;

    state = 11000001100;

    data = 11000001101;

    while(*state!=1){continue;}

    return *data;

}
```

B) char read\_kbd\_polling\_char(){

```
    char *data;

    int *state;

    state = 11000001100;

    data = 11000001101;

    if(*state==1){return *data;}

    else{return '\0';}

}
```

### **Ασκηση 13.6**

**(α)** (i) καθε διακοπη κοστιζει στον επεξεργαστη 2000 κυκλους ρολογιου.

(ii) για να κανουμε δειγματοληψια και στις 40 συσκευες θελουμε συνολικα  $200 \cdot 40 = 8000$  κυκλους.

Αν ο επεξεργατης κανει δειγματοληψια 1000 φορες το δευτερολεπτο σε ενα δευτερολεπτο θα εχουμε ξοδεψει  $1000 \cdot (2000 + 8000) = 10.000.000$  κυκλους ρολογιου. Δεδομενου οτι ο επεξεργαστης εκτελει 1.000.000.000 εντολες το δευτερολεπτο, οι κυκλοι δειγματοληψιας ειναι  $10.000.000 / 1.000.000.000 = 1\%$  (των συνολικων κυκλων)

**(β)** Καθε δευτερολεπτο ερχονται 20 νεες εισοδοι απο καθε συσκευη. Οποτε εχουμε συνολικα  $20 \cdot 40 = 800$  νεες εισοδοι απο ολες τις γραμμες. Καθε μια εισοδος κοστιζει 2000 κυκλους, επι 800 εισοδους κατα μεσο ορο, 1.600.000 κυκλοι ρολογιου. Το ποσοστο της υπολογιστικης του δυναμικης ειναι 0,16% και ειναι λιγοτερο απο του ερωτηματος α (1%) αρα συμφερει.

**(γ)** Πλεον καθε δευτερολεπτο ερχονται 500 νεες εισοδοι απο καθε συσκευη, αρα εχουμε  $40 \cdot 500 = 20000$  νεες εισοδους. Αυτες οι εισοδοι θα κοστισουν συνολικα  $20000 \cdot 2000 = 40.000.000$  κυκλους ρολογιου. Αυτοι οι κυκλοι ειναι το 4% των συνολικων κυκλων, οποτε εδω συμφερει η δειγματοληψια.

**(δ)** Με τον τρόπο β και γ δεν κινδυνεύουμε να χάσουμε κάποια εισοδο. Ο επεξεργαστής ξοδεύει 2000 κυκλους ρολογιου για καθε διακοπη, οποτε για τι 40 γραμμες θα χρειαστει 80.000 κυκλους ρολογιου (αρα 8ms). Για αυτον τον λογο ο επεξεργαστης δεν θα εχει τελειωσει με τις προηγουμενες διακοπες οποτε θα χασει την εισοδοαπο την γραμμη Α

**(ε)** Με ταχυτητα 32Mbit/s ο επεξεργαστης θα λαμβανει 100.000 εισοδους το δευτερολεπτο, αρα καθε δευτερολεπτο θα ξοδευει  $100.000 * 2.000 = 200.000.000$  κυκλους ρολογιου το δευτερολεπτο. Ετσι το ποσοστο υπολογιστικης δυναμης που καταναλωνει ανερχεται στο 20%. Με την μεθοδο της δειγματοληψιας απο την αλλη θα ξοδευαμε συνολικα 2200 κυκλους για καθε δειγματοληψια οποτε συνολικα 2.200.000 κυκλους (αρα το 0,0022% της υπολογιστικης δυναμης). Ξεκαθαρα λοιπον, σε αυτην την περιπτωση, η δειγματοληψια συμφερει

**(στ)** Με εισοδους της ταξης του 1Gbit/s θα εχουμε συνολικα 3.000.000 εισοδους το δευτερολεπτο, οποτε  $3.000.000 * 2.000 = 6.000.000.000$  κυλους ρολογιου το δευτερολεπτο. Μιας και ο επεξεργαστης μας εκτελει 1.000.000.000 κυκλους το δευτερολεπτο ειναι αδυνατο να διαχειρηστει τοσες εισοδους μαζι.

### **Ασκηση 13.7**

**(α)** Γνωριζουμε οτι ο ρυθμος αντιγραφης ειναι 50MBps = 400 Mbps ( = 4Bytes/80cc = 1Byte/80ns). Για τους δυο δισκους θα χρειαστει 160Mbits/s(=2\*10MB/s) και για το δικτυο 100Mbits/s, οποτε συνολικα 260Mbps (αρα 65% της ισχυς του)

**(β) (i)** Με το bus ταχυτητας 312,5Mbps (=1B/3,2ns) και με 19,5MB πακετα(16Bytes ανα πακετο), εχουμε 624Mbps (=78MBps=312-(19,5\*12))

**(ii)** Αντιστοιχα στελνοντας 4M πακετα των 64B (χωρις τα headers) θα εχουμε  $312,5 - (4 * 12) = 264,5$ MBps ==2116Mbps

**(γ)** Οπως και στο ερωτημα **(α)** θαχρειαστει συνολικα 260Mbps (=160Mbps δισκου + 100Mbps δικτυου) οποτε το 10% ισχυς της αρτηριας. Αυτην την φορα μεταφερουμε ολοκληρα πακετα αντι για ξεχωριστες λεξεις