

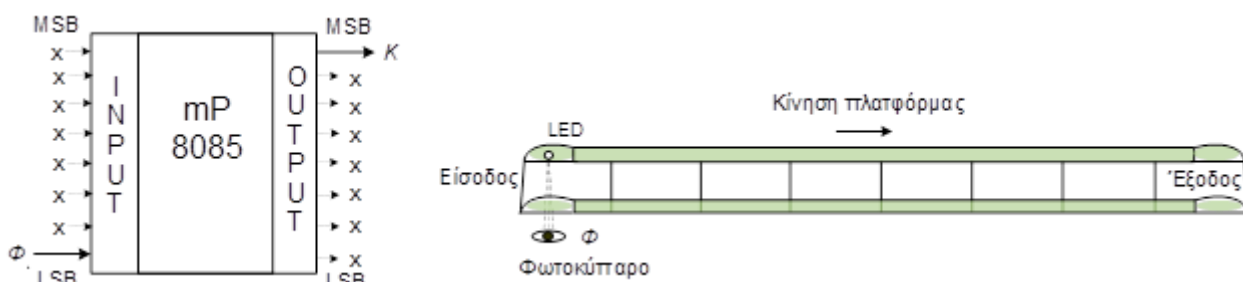
ΓΡΑΠΤΗ ΕΞΕΤΑΣΗ ΣΤΟ ΜΑΘΗΜΑ "Συστήματα Μικροϋπολογιστών"

(ΘΕΜΑ 1^ο – ΣΥΝΟΛΟ 3.5 Μονάδες)

Έναρξη 11:30 - ΔΙΑΡΚΕΙΑ 50' + 10' Παράδοση: 12:30'

ΟΝΟΜΑΤΕΠΩΝΥΜΟ:

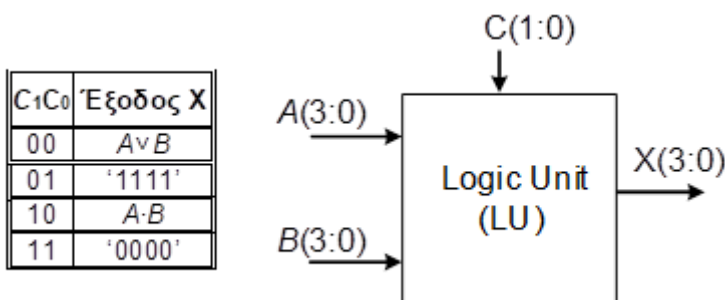
ΘΕΜΑ 1α: (2 ΜΟΝΑΔΕΣ): Δίνεται μΥ-Σ που διαθέτει δύο 8-bit θύρες: μία εισόδου (διεύθ. 20^{HEX}) και μία εξόδου (διεύθ. 30^{HEX}). Να γραφεί πρόγραμμα assembly σε 8085 που να υλοποιηθεί ένα σύστημα οδήγησης κυλιόμενης πλατφόρμας μονής κατεύθυνσης η οποία να ενεργοποιείται από το φωτοκύτταρο Φ . Συγκεκριμένα, αν ένας επιβάτης εισέρχεται στην πλατφόρμα, όταν είναι ακίνητη, διακόπτει δέσμη φωτός (γίνεται $\Phi=0$) και τότε τίθεται σε κίνηση η πλατφόρμα με το σήμα εξόδου K (για $K=1$ έχουμε κίνηση). Η κίνηση να σταματά ~10 sec μετά την τελευταία διακοπή του φωτοκυττάρου Φ (χρόνος ικανός για να αδειάσει η πλατφόρμα από επιβάτες). Μπορείτε να κάνετε χρήση της ρουτίνας χρονοκαθυστέρησης DSEC των 50 msec.



ΘΕΜΑ 1β: (0.8 ΜΟΝΑΔΕΣ): Απαντήστε στα παρακάτω ερωτήματα (σύντομα και αιτιολογημένα):

- (i) Δώστε τη μακροεντολή MIN που μετακινεί τον ελάχιστον των καταχωρητών B, D, E στον καταχωρητή A , χωρίς να επηρεάζεται η τιμή των καταχωρητών (πλην φυσικά του A). (0.4 ΜΟΝΑΔΕΣ)
- (ii) Εξηγήστε τη λειτουργική διαφορά των καθυστερήσεων που προκαλούνται μέσω ρουτινών χρονοκαθυστέρησης και μέσω μετρητών-χρονιστών (πλεονεκτήματα, μειονεκτήματα). (0.2 ΜΟΝΑΔΕΣ)
- (iii) Πώς λειτουργούν οι εντολές σχετικού άλματος και κλήσης ρουτίνας; Εξηγήστε την χρησιμότητά τους. (0.2 ΜΟΝΑΔΕΣ)

ΘΕΜΑ 1γ: (0.7 ΜΟΝΑΔΕΣ): Να δοθεί το κύκλωμα (σχηματικό διάγραμμα) και η δομική περιγραφή σε Verilog της μονάδας LU που η λειτουργία της φαίνεται στο διπλανό πίνακα και σχήμα. Μπορείτε να κάνετε χρήση των βασικών πυλών: $XOR(x,a,b)$, $OR(x,a,b)$, $AND(x,a,b)$ και $INV(x,a)$ θεωρώντας τις μεταβλητές a και b ως εισόδους. Το σύμβολο 'v' δηλώνει την πράξη OR ενώ το '.' την πράξη AND. Υποθέτουμε ότι οι μεταβλητές A, B και X είναι των 4-bit. Η λέξη ελέγχου $C(1:0)$ είναι των 2-bit. Επίσης να δοθεί η περιγραφή Verilog του ίδιου κυκλώματος σε μορφή ροής δεδομένων ή σε μοντελοποίηση συμπεριφοράς.



ΓΡΑΠΤΗ ΕΞΕΤΑΣΗ ΣΤΟ ΜΑΘΗΜΑ "Συστήματα Μικροϋπολογιστών"

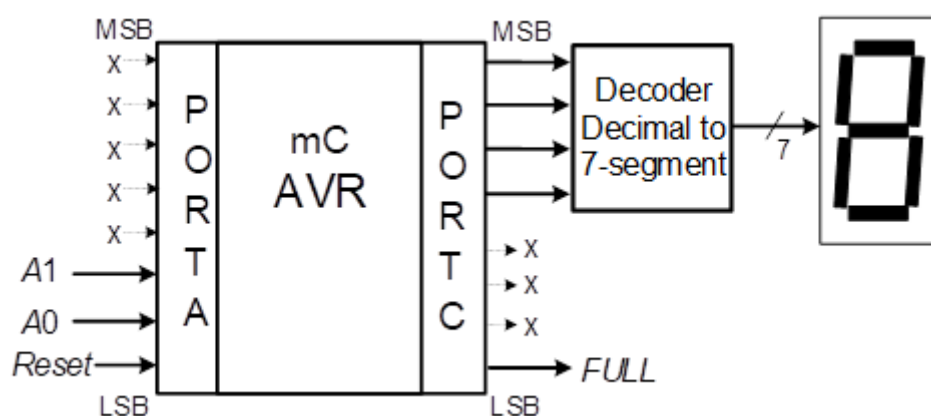
(ΘΕΜΑ 2^ο – ΣΥΝΟΛΟ 4.5 Μονάδες)

Έναρξη 12:30 - ΔΙΑΡΚΕΙΑ 60' + 10' Παράδοση: 13:40'

ΟΝΟΜΑΤΕΠΩΝΥΜΟ:

ΘΕΜΑ 2ο: (4.5 ΜΟΝΑΔΕΣ): Σε ένα μικροελεγκτή AVR Mega16 που φαίνεται στο σχήμα, να υλοποιηθεί ένα σύστημα απεικόνισης των ελεύθερων θέσεων ενός parking. Υποθέτουμε ότι υπάρχει χωριστή είσοδος και έξοδος, εφοδιασμένες η κάθε μία με δέσμη φωτός και φωτοκύτταρα A0 και A1 αντίστοιχα. Κάθε όχημα που περνάει, διακόπτει τη δέσμη φωτός (το φωτοκύτταρο τότε δίνει 0) και στη συνέχεια αυτή επανέρχεται. Για την απλούστευση της λύσης υποθέτουμε ότι αποκλείεται η περίπτωση να έχουμε ταυτόχρονα είσοδο και έξοδο οχήματος. Το σύστημα διαθέτει είσοδο *Reset* που όταν τεθεί στο '1', υποθέτοντας ότι τότε το parking είναι άδειο, να αρχικοποιεί τις ελεύθερες θέσεις στις 30. Επίσης στην εκκίνηση του συστήματος το parking να θεωρηθεί άδειο. Όταν ο αριθμός των ελεύθερων θέσεων είναι <10, αυτός να απεικονίζεται στο 7-segment display, αλλιώς να απεικονίζεται ο αριθμός 9 και να ανάβει το led (θετικής λογικής) με την ένδειξη FULL. Υποθέτουμε ότι αν το parking γεμίσει αποκλείεται η είσοδος άλλων οχημάτων. Για τη δική σας διευκόλυνση φτιάξτε ένα πρόχειρο διάγραμμα ροής. Δώστε το πρόγραμμα υλοποίησης του παραπάνω συστήματος σε assembly και σε C.

(Assembly: 2.5 ΜΟΝΑΔΕΣ και C: 2 ΜΟΝΑΔΕΣ)



ΓΡΑΠΤΗ ΕΞΕΤΑΣΗ ΣΤΟ ΜΑΘΗΜΑ "Συστήματα Μικροϋπολογιστών"

(ΘΕΜΑ 3^ο – ΣΥΝΟΛΟ 2 Μονάδες)

Έναρξη 13:40' - ΔΙΑΡΚΕΙΑ 30' + 10' Παράδοση: 14:20'

ΟΝΟΜΑΤΕΠΩΝΥΜΟ:

ΘΕΜΑ 3ο: (2 ΜΟΝΑΔΕΣ): Σε ένα προσωπικό υπολογιστή, να γραφεί πρόγραμμα σε Assembly με 80x86 που να δέχεται από το πληκτρολόγιο τέσσερις (4) δεκαεξαδικούς αριθμούς (H_3, H_2, H_1, H_0 με τη σειρά αυτή) για να αποτελέσουν δυο μονοψήφιους και ένα διψήφιο δεκαεξαδικό αριθμό και να κάνει τον εξής υπολογισμό:

$P = H_3 + H_2 \times (H_1 \times 10 + H_0)$. Το πρόγραμμα τυπώνει στην οθόνη τα μηνύματα εισόδου και τους εισαγόμενους αριθμούς. Όταν συμπληρωθούν 4 έγκυροι δεκαεξαδικοί αριθμοί να αναμένει τον χαρακτήρα 'H' και μετά να τυπώνει το αποτέλεσμα σε δεκαδική μορφή 3 ψηφίων αν δεν υπάρχει υπερχείλιση (δηλ. < 1000). Αν όμως υπάρχει υπερχείλιση (> 999) τότε να τυπώνει το μήνυμα OVERF, αυστηρά όπως φαίνεται παρακάτω:

DOSE 1ο ARITHMO = D

DOSE 2ο ARITHMO = 7

DOSE 3ο ARITHMO = 5C

APOTELESMA = 657 ή APOTELESMA = YPERX

Να θεωρήσετε δεδομένες τις μακροεντολές (σελ. 361-2, 373) του βιβλίου και μπορείτε να κάνετε χρήση των ρουτινών DEC_KEYB και PRINT_HEX χωρίς να συμπεριλάβετε τον κώδικά τους. Για την διευκόλυνσή σας, δίνονται οι πρώτες εντολές που αποτελούν τον 'σκελετό' του ζητούμενου προγράμματος.

ΑΠΑΝΤΗΣΗ

INCLUDE MACROS

DATA_SEG SEGMENT

MSG1 DB 0AH,0DH, 'DOSE 1ο ARITHMO = \$'

MSG2 DB 0AH,0DH, 'DOSE 2ο ARITHMO = \$'

MSG3 DB 0AH,0DH, 'DOSE 3ο ARITHMO = \$'

MSG4 DB 0AH,0DH, 'APOTELESMA = \$'

DATA_SEG ENDS

CODE_SEG SEGMENT

ASSUME CS:CODE_SEG, DS:DATA_SEG

MAIN PROC FAR

MOV AX, DATA_SEG

MOV DS, AX

...