

Εργασία 2

ι. Επανάληψη – Επιλογή

Εκτελέστε το παρακάτω πρόγραμμα και συμπληρώστε σε έναν πίνακα τις τιμές των καταχωρητών R0-R2 μετά την εκτέλεση των γραμμών @E0-@E3 μέχρι και τον τερματισμό του προγράμματος.

.arm

.text

.global main

main:

STMDB R13!, {R0-R12, R14}

MOV R2, #0x00

LDR R0, =Values @E0

LDRB R1, [R0] @E1

LOOP:

CMP R1, #0xc2

MOVEQ R2, R1 @E2

ADD R1, R1, #0x1 @E3

BLT LOOP

LDMIA R13!, {R0-R12, PC}

.data

Values:

.byte 0xc1, 0xb2, 0xf3, 0xf4

ii.Αθροιση bytes

Δίδονται οι δύο παρακάτω πίνακες Α και Β. Κάθε ένας τους περιέχει 16 στοιχεία, όπου κάθε στοιχείο είναι ένας δεκαδικός αριθμός που απαιτεί για την αναπαράστασή του στο δυαδικό σύστημα 8 δυαδικά ψηφία. Αναπτύξτε πρόγραμμα σε συμβολική γλώσσα που να προσθέτει τα αντίστοιχα στοιχεία των δύο αυτών πινάκων (το στοιχείο της 1ης γραμμής του πίνακα Α προστίθεται με αυτό της 1ης γραμμής του πίνακα Β, κοκ.) και να αποθηκεύει τα 16 αποτελέσματα σε ένα νέο πίνακα, έστω Γ, του οποίου τα στοιχεία έχουν εύρος 8 δυαδικά ψηφία το καθένα. Η μεθοδολογία που θα ακολουθήσετε για την ανάπτυξη του προγράμματός σας θα πρέπει να χρησιμοποιεί ετικέτες και άλματα υπό συνθήκη. Συμπληρώστε τα αθροίσματα που προέκυψαν στον παρακάτω πίνακα στο δεκαεξαδικό, και κάνοντας τη μετατροπή, και στο δεκαδικό. Στη τελευταία στήλη του πίνακα σημειώστε με «X» κάθε περίπτωση που το αποτέλεσμα διαφέρει από το αναμενόμενο. Προσπαθείστε να εξηγήσετε τα μη αναμενόμενα αποτελέσματα.

| byte | Πίνακας Α | Πίνακας Β | Πίνακας Γ | | Μη αναμενόμενο |
|------|--------------|--------------|-------------|----------|-------------------|
| | | | Δεκαεξαδικό | Δεκαδικό | |
| 0 | 32 | 19 | | | |
| 1 | 127 | 1 | | | |
| 2 | 254 | 18 | | | |
| 3 | 57 | 89 | | | |
| 4 | 22 | 90 | | | |
| 5 | 111 | 112 | | | |
| 6 | 48 | 89 | | | |
| 7 | 11 | 32 | | | |
| 8 | 87 | 23 | | | |
| 9 | 45 | 98 | | | |
| 10 | 114 | 67 | | | |
| 11 | 45 | 83 | | | |
| 12 | 66 | 146 | | | |
| 13 | 23 | 140 | | | |
| 14 | 134 | 200 | | | |
| 15 | 168 | 67 | | | |

iii. Αθροιση halfwords

| byte | Πίνακας Α | Πίνακας Β | Πίνακας Γ |
|------|-----------|-----------|-----------|
| 0 | 32 | 19 | |
| 1 | 127 | 1 | |
| 2 | 254 | 18 | |
| 3 | 57 | 89 | |
| 4 | 22 | 90 | |
| 5 | 111 | 112 | |
| 6 | 48 | 89 | |
| 7 | 12 | 22 | |
| 8 | 87 | 23 | |
| 9 | 45 | 98 | |
| 10 | 114 | 67 | |
| 11 | 45 | 83 | |
| 12 | 66 | 146 | |
| 13 | 23 | 140 | |
| 14 | 134 | 200 | |
| 15 | 168 | 67 | |

Στο δεύτερο μέρος πρέπει να εκτελέσετε τις αντίστοιχες προσθέσεις, θεωρώντας πως οι πίνακές μας αποτελούνται από 8 στοιχεία - αριθμούς των 16 δυαδικών ψηφίων ο καθένας. Τα αποτελέσματα της κάθε πρόσθεσης θα πρέπει να είναι επίσης εύρους 16 δυαδικών ψηφίων. Για παράδειγμα, η πρώτη πρόσθεση θα γίνει ανάμεσα στους αριθμούς ($127 * 256 + 32 = 32544$) και ($1 * 256 + 19 = 275$). Συμπληρώστε τα αποτελέσματα που προκύπτουν στο δεκαδικό. Θυμηθείτε πως η αρχιτεκτονική μας είναι little endian, το οποίο σημαίνει ότι το λιγότερο σημαντικό byte ενός halfword είναι τοποθετημένο στη χαμηλότερη διεύθυνση μνήμης, ενώ το περισσότερο σημαντικό byte είναι τοποθετημένο στην υψηλότερη. Στην περίπτωση μας, τα bytes στις θέσεις 1,3,5,... είναι τα περισσότερο σημαντικά, ενώ τα bytes στις θέσεις 0,2,4,... είναι τα λιγότερα σημαντικά.

iv. Αθροιση words

| byte | Πίνακας Α | Πίνακας Β | Πίνακας Γ |
|------|-----------|-----------|-----------|
| 0 | 32 | 19 | |
| 1 | 127 | 1 | |
| 2 | 254 | 18 | |
| 3 | 57 | 89 | |
| 4 | 22 | 90 | |
| 5 | 111 | 112 | |
| 6 | 48 | 89 | |
| 7 | 12 | 22 | |
| 8 | 87 | 23 | |
| 9 | 45 | 98 | |
| 10 | 114 | 67 | |
| 11 | 45 | 83 | |
| 12 | 66 | 146 | |
| 13 | 23 | 140 | |
| 14 | 134 | 200 | |
| 15 | 168 | 67 | |

Στο τρίτο μέρος πρέπει να γίνουν οι αθροίσεις θεωρώντας πως τα στοιχεία των πινάκων Α και Β έχουν εύρος 32 δυαδικά ψηφία, όπως επίσης και τα Σητούμενα αθροίσματα. Και εδώ θα πρέπει να λάβετε υπόψη σας την little endian αρχιτεκτονική και τη σειρά των bytes.

v. Αθροιση longwords

| byte | Πίνακας Α | Πίνακας Β | Πίνακας Γ |
|------|-----------|-----------|-----------|
| 0 | 32 | 19 | |
| 1 | 127 | 1 | |
| 2 | 254 | 18 | |
| 3 | 57 | 89 | |
| 4 | 22 | 90 | |
| 5 | 111 | 112 | |
| 6 | 48 | 89 | |
| 7 | 12 | 22 | |
| 8 | 87 | 23 | |
| 9 | 45 | 98 | |
| 10 | 114 | 67 | |
| 11 | 45 | 83 | |
| 12 | 66 | 146 | |
| 13 | 23 | 140 | |
| 14 | 134 | 200 | |
| 15 | 168 | 67 | |

Στο τελευταίο μέρος θα θεωρήσουμε πως οι πίνακές μας έχουν ένα μόνο στοιχείο εύρους 16 bytes. Το αποτέλεσμα της άθροισης αυτών των δύο στοιχείων (το στοιχείο δηλαδή του πίνακα Γ) θα πρέπει επίσης να έχει εύρος 16 bytes. Επειδή η αρχιτεκτονική της ΑΛΜ του επεξεργαστή μας δεν υποστηρίζει πράξεις επί ποσοτήτων μεγαλύτερων από 32 δυαδικά ψηφία θα πρέπει να κατασκευάσουμε ένα πρόγραμμα το οποίο να υλοποιεί αυτή

την εξειδικευμένη πρόσθεση. Ξεκινάμε με την πρόσθεση των λιγότερο σημαντικών words, αποθηκεύουμε το κρατούμενο της πράξης στη σημαία Carry του καταχωρητή κατάστασης και το χρησιμοποιούμε στην πρόσθεση των δύο αμέσως σημαντικότερων words. Η διαδικασία αυτή θα πρέπει να επαναληφθεί 4 φορές, μέχρι να προσθέσουμε και τις περισσότερες σημαντικές words των αριθμών μας. Πόσες προσθέσεις θα χρειαζόμασταν αν εκτελούσαμε προσθέσεις bytes και όχι words;

