



ΠΟΛΥΤΕΧΝΕΙΟ
ΚΡΗΤΗΣ

Εργαστηριακή Άσκηση 10^η

Κάνοντας υπολογισμούς με «μικρούς» μικροελεγκτές - πολλαπλασιασμός 3X3
πινάκων float και int

Σωτήριος Μιχαήλ
2015030140

**ΕΝΣΩΜΑΤΩΜΕΝΑ
ΣΥΣΤΗΜΑΤΑ
ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ**

Εισαγωγή

Σκοπός της εργαστηριακής άσκησης είναι η υλοποίηση κώδικα που εκτελεί απλές πράξεις μεταξύ πινάκων 3x3, σε αυτή την άσκηση θα υπάρξει υλοποίηση μόνο του πολλαπλασιασμού.

$$A \cdot B = C$$

Τα στοιχεία των A, B και C πινάκων θα είναι είτε ακέραιοι αριθμοί (int) είτε αριθμοί κινητής υποδιαστολής (float). Στόχος είναι να αξιολογήσουμε το αντίκτυπο σε πόρους του μικροελεγκτή που θα έχει μία υλοποίηση αριθμητικής κινητής υποδιαστολής, σε σχέση με την πιο απλή αριθμητική ακεραίων αριθμών.

Υλοποίηση

Αρχικά, τα στοιχεία των πινάκων καθώς και η βοηθητική μεταβλητή δηλώνονται ως “element_t”, έτσι ώστε εύκολα να αλλάζει ο κώδικας και να έχουμε είτε αριθμητική ακεραίων είτε αριθμών κινητής υποδιαστολής. Οι πίνακες δηλώθηκαν με το attribute “volatile” έτσι ώστε να μην αφαιρεθούν από τις βελτιστοποιήσεις του μεταγλωττιστή avr-gcc, καθώς δεν έχουμε έξοδο για το αποτέλεσμα του πολλαπλασιασμού.

Αρχικοποίηση

Οι πίνακες A και B οι οποίοι πολλαπλασιάζονται, αρχικοποιούνται με τυχαίους αριθμούς, οι οποίοι μετατρέπονται είτε σε αριθμούς κινητής υποδιαστολής, είτε σε ακέραιους αριθμούς, ανάλογα με τον δηλωμένο τύπο του element_t.

Θέσεις μνήμης

Οι πίνακες A, B και C βρίσκονται στη μνήμη SRAM του μικροελεγκτή, στη περιοχή “.data”. Μέσω του avr-nm, μπορούμε να έχουμε ένα χάρτη της μνήμης, και να δούμε σε ποιες ακριβώς διευθύνσεις βρίσκονται οι πίνακες, καθώς και άλλες μεταβλητές.

Με ακέραιους αριθμούς		Με αριθμούς κινητής υποδιαστολής	
Διεύθυνση μνήμης	Στοιχείο	Διεύθυνση μνήμης	Στοιχείο
00800060	c	00800060	c
00800060	__DATA_REGION_ORIGIN	00800060	__DATA_REGION_ORIGIN
00800060	__data_start	00800060	__data_start
00800072	b	00800084	b
00800084	a	008000a8	a
00800096	next	008000cc	next
0080009a	__data_end	008000d0	__data_end
0080009a	__edata	008000d0	__edata
0080009a	__end	008000d0	__end
00810000	__eeprom_end	00810000	__eeprom_end

Μεταβλητές

Μεταβλητή	Χρήση
A, B, C sum	Οι πίνακες του πολλαπλασιασμού Βοηθητική μεταβλητή

Χρήση πόρων

Μέσω του avr-size και του προσομοιωτή του μικροελεγκτή ATmega16 του Atmel Studio 7.0, μπορούμε να προσδιορίσουμε το πόσους πόρους χρησιμοποιεί ο κώδικας που υλοποιήθηκε σε αυτό το εργαστήριο.

Ο ATmega16 έχει:

- 16 kbyte Flash μνήμη προγράμματος
- 512 byte EEPROM
- 1 kbyte εσωτερική SRAM

Για αριθμητική ακεραίων αριθμών, η χρήση πόρων είναι:

- Μνήμη προγράμματος (.text + .data + .bootloader): 876 bytes (5.3% full)
- Μνήμη δεδομένων (.data + .bss + .noinit): 58 bytes (5.7% full)
- Κύκλοι εκτέλεσης πολλαπλασιασμού των δύο πινάκων: 1881 κύκλοι, 1.88 ms στα 1 MHz

Για αριθμητική αριθμών κινητής υποδιαστολής, η χρήση πόρων είναι:

- Μνήμη προγράμματος (.text + .data + .bootloader): 1708 bytes (10.4% full)
- Μνήμη δεδομένων (.data + .bss + .noinit): 112 bytes (10.9% full)
- Κύκλοι εκτέλεσης πολλαπλασιασμού των δύο πινάκων: 8782 κύκλοι, 8.78 ms στα 1 MHz

Συμπέρασμα άσκησης

Παρατηρείται πως η υλοποίηση με αριθμητική αριθμών κινητής υποδιαστολής είναι ιδιαίτερα δαπανηρή, καθώς μόνον οι τρεις πίνακες 3x3 και ο μηχανισμός πολλαπλασιασμού τους καταναλώνουν το 1/10 της διαθέσιμης μνήμης του μικροελεγκτή, τους διπλάσιους πόρους μνήμης της υλοποίησης με αριθμητική ακεραίων. Επίσης, ο χρόνος εκτέλεσης είναι 4.6 φορές μεγαλύτερος του χρόνου εκτέλεσης του πολλαπλασιασμού με ακέραιους αριθμούς. Επομένως, η αριθμητική κινητής υποδιαστολής πρέπει να χρησιμοποιείται μόνον όπου κρίνεται απολύτως αναγκαία.