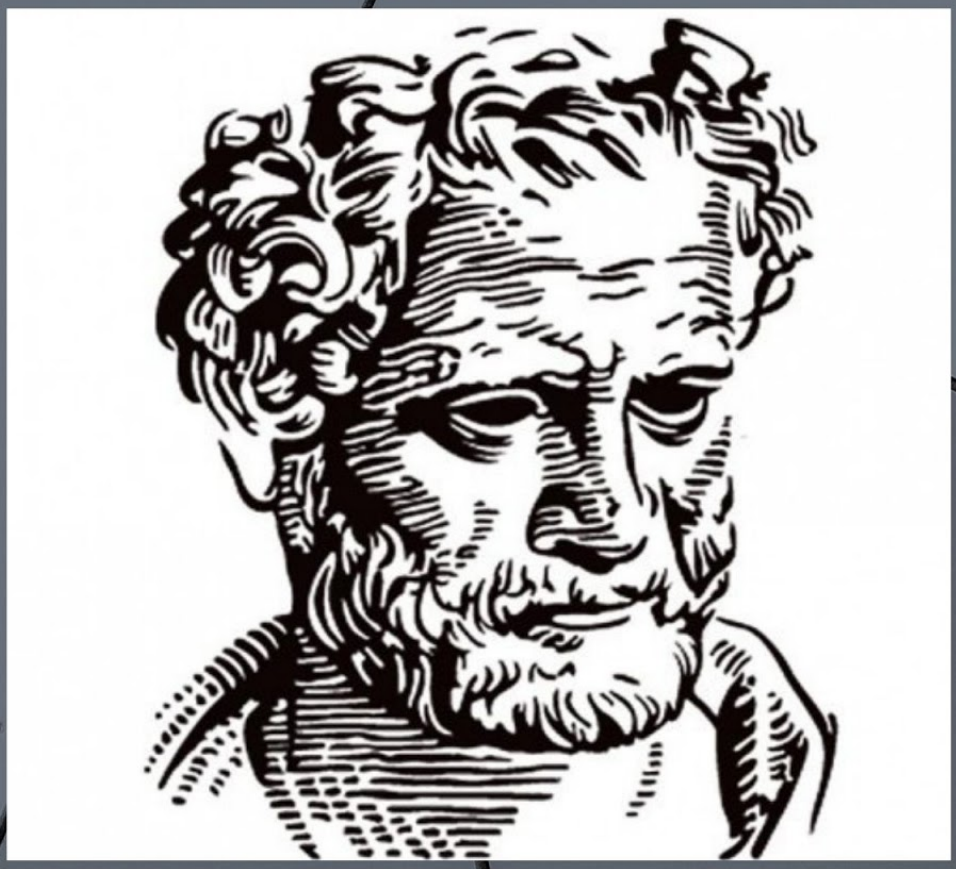
ΙΑΝΟΥΑΡΙΟΣ 2021

ΕΡΓΑΣΙΑ 3

Σχεδιασμός Ενσωματωμένων Συστημάτων

ΟΜΑΔΑ 39



Υπεύθυνος καθηγητής : Συρακούλης Γεώργιος

Φοιτητές : Κεφαλάς Γεώργιος (57406)
Τσίτσος Δημήτριος (57407)

1. Εισαγωγή

Στα πλαίσια του τρίτου μέρους της εργασίας για το μάθημα του Σχεδιασμού Ενσωματωμένων Συστημάτων εξετάσαμε την επαναχρησιμοποίηση της μνήμης. Προσπαθήσαμε να βελτιστοποιήσουμε τον αριθμο προσπελάσεων στη μνήμη, έτσι ώστε ο χρόνος εκτέλεσης του προγράμματος να γίνει ο μικρότερος δυνατός. Επιπλέον, προσομοιώσαμε τον αλγόριθμό μας στον Armulator του επεξεργαστή ARM και παρακάτω παρατίθενται τα αποτελέσματα μαζί με τα συμπεράσματα μας.

2. Περιγραφή προβλήματος

Αρχικά, για την υλοποίηση της επαναχρησιμοποίησης της μνήμης εισάγαμε στον αλγόριθμο που είχαμε ήδη αναπτύξει 2 νέες μεταβλητές (buffers) με τη μορφή πινάκων οι οποίες προσπελαύνουν την γρήγορη μνήμη SRAM. Αυτές χρησιμοποιούνται για μεταβλητές οι οποίες εκτελούνται πολλες φορές σε κάθε επανάληψη κατα τη διάρκεια εκτέλεσης του αλγόριθμου μας. Με αυτό τον τρόπο, γλυτώνουμε χρόνο και έξτρα κύκλους που θα χρειαζόμασταν για την προσωρινή αποθήκευση των μεταβλητών αυτών στη μνήμη μας. Επιπλέον, διατηρούμε την ίδια ιεραρχία μνήμης με το δεύτερο μέρος της εργασίας, καθώς αποθηκεύουμε τις τιμές κάποιων μεταβλητών στη RAM ενώ κάποιες άλλες στην SRAM.

```
00000000 00080000 ROM 4 R 1/1 1/1
00080000 08000000 RAM 4 RW 250/50 250/50
08080000 00008000 SRAM 4 RW 1/1 1/1
```

Το αρχείο memory.map

```
ROM 0x0 0x08088000
{
ROM 0x0 0x80000
{
*.o ( +RO )
}
RAM 0x80000 0x08000000
{
* ( ram )
* ( +ZI )
}
SRAM 0x08080000 0x00008000
{
* ( sram )
}

}
```

Το αρχείο scatter.

Σε αντίθεση με το 3ο εργαστήριο του μαθήματος προσεγγίσαμε το πρόβλημα επαναχρησιμοποίησης της μνήμης διαφορετικά. Η βασική διαφορά του αλγόριθμου μας ήταν ότι δεν χρησιμοποιούνται convolutional παράθυρα, δεν ήταν δηλαδή εφικτή η χρήση των buffers με την μέθοδο ολίσθησης των παραθύρων. Γι'αυτό το λόγο, επιλύσαμε το τρίτο μέρος της εργασίας όπως περιγράφεται και παραπάνω,

3. Υλοποίηση κώδικα

Τα παρακάτω είναι screenshots με τα αποτελέσματα εκτέλεσης αρχικά με τη μέθοδο decomposition και έπειτα με τη μέθοδο averaging για διάφορες μεθόδους βελτιστοποίησης επαναλήψεων.

- **Decomposition**

(Αποτελέσματα make)

```
================================================================================

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      3288         60          0    2032836       7516   Object Totals
     13904        314          0        300       6032   Library Totals
================================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     17192        374          0    2033136      13548   Grand Totals
================================================================================

    Total RO  Size(Code + RO Data)                17566 (  17.15kB)
    Total RW  Size(RW Data + ZI Data)           2033136 (1985.48kB)
    Total ROM Size(Code + RO Data + RW Data)      17566 (  17.15kB)
================================================================================
```

(Αποτελέσματα statistics)

Debugger Internals

Internal Variables | Statistics

Referenc...	Instruct...	Core_Cycles	S_Cycles	N_Cycles	I_Cycles	C_Cycles	Wait_States	Total	True_Idl...
$statistics	247921415	359904387	274114598	66714976	70387519	0	73449834	484666927	28229012

- **Averaging**

(Αποτελέσματα make)

```
==============================================================================

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      3288         60          0    2032836       7516   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     17192        374          0    2033136      13548   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                17566 (  17.15kB)
    Total RW  Size(RW Data + ZI Data)           2033136 (1985.48kB)
    Total ROM Size(Code + RO Data + RW Data)      17566 (  17.15kB)
==============================================================================
```

(Αποτελέσματα statistics)

Debugger Internals

Internal Variables | Statistics

Referenc...	Instruct...	Core_Cycles	S_Cycles	N_Cycles	I_Cycles	C_Cycles	Wait_States	Total	True_Idl...
$statistics	253942742	366099438	280631128	66655738	70159013	0	73449834	490895713	28250180

- Loop Unrolling **(Decomposition)**

(Αποτελέσματα make)

```
========================================================================

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      4424         60          0    2032836       7696   Object Totals
     13904        314          0        300       6032   Library Totals
========================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     18328        374          0    2033136      13728   Grand Totals
========================================================================

    Total RO  Size(Code + RO Data)                18702 (  18.26kB)
    Total RW  Size(RW Data + ZI Data)           2033136 (1985.48kB)
    Total ROM Size(Code + RO Data + RW Data)      18702 (  18.26kB)
========================================================================
```

(Αποτελέσματα statistics)

Debugger Internals

Internal Variables | Statistics

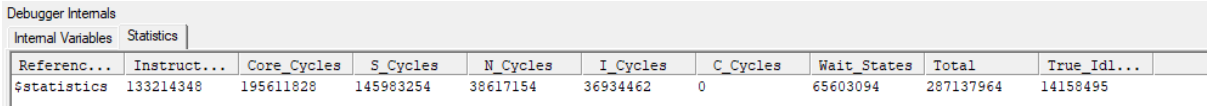
Referenc...	Instruct...	Core_Cycles	S_Cycles	N_Cycles	I_Cycles	C_Cycles	Wait_States	Total	True_Idl...
$statistics	130364980	193406420	142946095	39019876	37361169	0	65603094	284930234	14155849

- Loop Unrolling **(Averaging)**

(Αποτελέσματα make)

```
========================================================================

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      4412         60          0    2032836       7660   Object Totals
     13904        314          0        300       6032   Library Totals
========================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     18316        374          0    2033136      13692   Grand Totals
========================================================================

    Total RO  Size(Code + RO Data)                18690 (  18.25kB)
    Total RW  Size(RW Data + ZI Data)           2033136 (1985.48kB)
    Total ROM Size(Code + RO Data + RW Data)      18690 (  18.25kB)
========================================================================
```

(Αποτελέσματα statistics)



- Loop Unrolling & Interchange **(Decomposition)**

(Αποτελέσματα make)

```
========================================================================

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      4400         60          0    2032836       7696   Object Totals
     13904        314          0        300       6032   Library Totals
========================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     18304        374          0    2033136      13728   Grand Totals
========================================================================

    Total RO  Size(Code + RO Data)                18678 (  18.24kB)
    Total RW  Size(RW Data + ZI Data)           2033136 (1985.48kB)
    Total ROM Size(Code + RO Data + RW Data)      18678 (  18.24kB)
========================================================================
```

(Αποτελέσματα statistics)

Debugger Internals

Internal Variables | Statistics

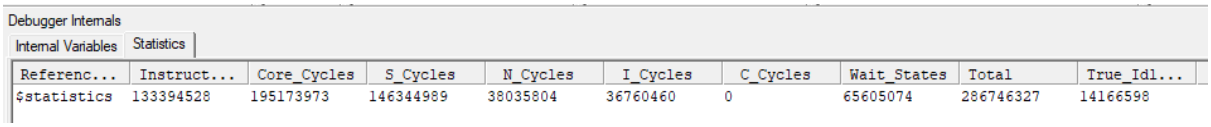
Referenc...	Instruct...	Core_Cycles	S_Cycles	N_Cycles	I_Cycles	C_Cycles	Wait_States	Total	True_Idl...
$statistics	130545160	193186365	143307830	38547426	37296067	0	65605074	284756397	14163952

- Loop Unrolling & Interchange **(Averaging)**

(Αποτελέσματα make)

```
========================================================================

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      4388         60          0    2032836       7660   Object Totals
     13904        314          0        300       6032   Library Totals
========================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     18292        374          0    2033136      13692   Grand Totals
========================================================================

    Total RO  Size(Code + RO Data)                18666 (  18.23kB)
    Total RW  Size(RW Data + ZI Data)           2033136 (1985.48kB)
    Total ROM Size(Code + RO Data + RW Data)      18666 (  18.23kB)
========================================================================
```

(Αποτελέσματα statistics)


Debugger Internals

Internal Variables | Statistics

Referenc...	Instruct...	Core_Cycles	S_Cycles	N_Cycles	I_Cycles	C_Cycles	Wait_States	Total	True_Idl...
$statistics	133394528	195173973	146344989	38035804	36760460	0	65605074	286746327	14166598

4. Συμπεράσματα

Αφου εκτελέσαμε τα αποτελέσματα βρήκαμε μια βελτίωση σε σύγκριση με τα προηγούμενα αποτελέσματα μας. Η βελτίωση είναι της τάξης του 5-6% καθώς γλυτώνουμε χρόνο και προσπαλάσεις στη RAM η οποία είναι πιο αργή σε σχέση με την SRAM που χρησιμοποιούμε σε αυτό το μέρος της εργασίας. Επιπλέον, παρατηρούμε ότι οι διάφορες μέθοδοι βελτιστοποίησης *(loop unrolling & loop interchange)* των επαναλήψεων βοηθούν στο να έχουμε λιγότερους κύκλους. Όπως φαίνεται και στα δεδομένα των παραπάνω εκτελέσεων, ο συνδυασμός των μεθόδων unrolling & interchange έχει ως αποτέλεσμα την ταχύτερη εκτέλεση του αλγορίθμου. Τελικά, συμπεραίνουμε ότι δίνοντας ιδιαίτερη προσοχή στην ανάπτυξη του αλγορίθμου μας ως προς τη διαχείριση των μεταβλητών μας, της μνήμης μας αλλά και τις μεθόδους επίλυσης του προβλήματος μας έχει σημαντικές επιπτώσεις στην απόδοση που έχουμε.