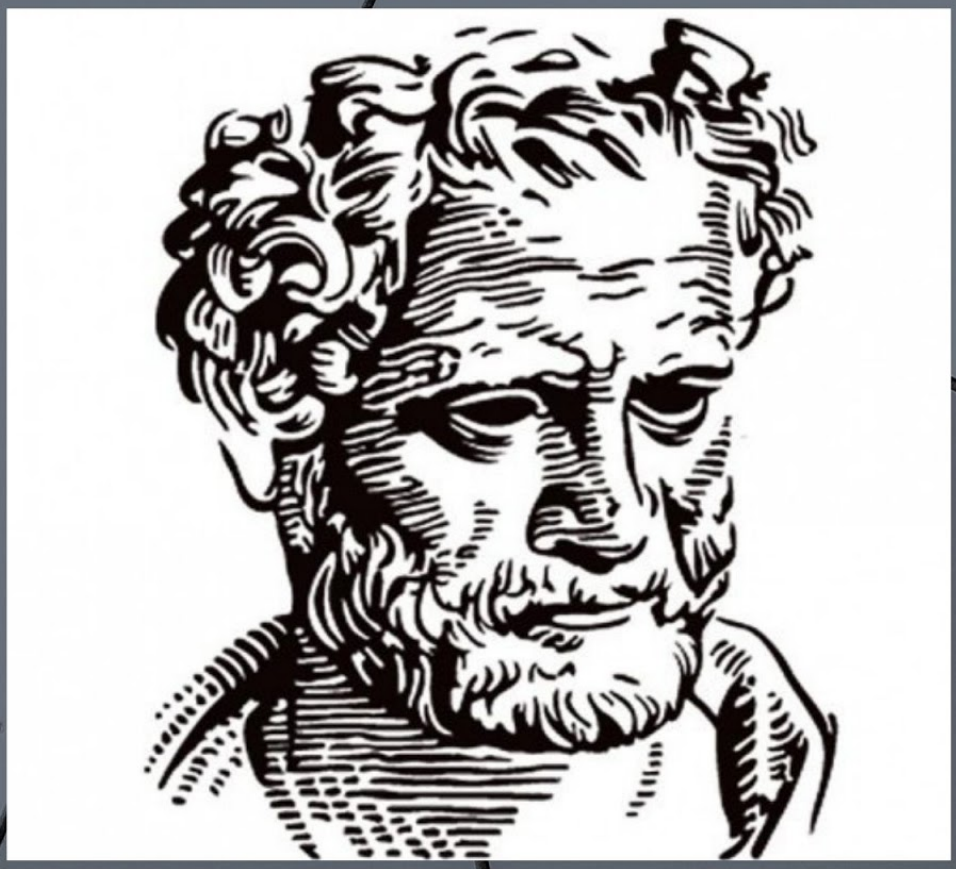
ΔΕΚΕΜΒΡΙΟΣ 2020

ΕΡΓΑΣΙΑ 2

Σχεδιασμός Ενσωματωμένων Συστημάτων

ΟΜΑΔΑ 39



Υπεύθυνος καθηγητής : Συρακούλης Γεώργιος

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

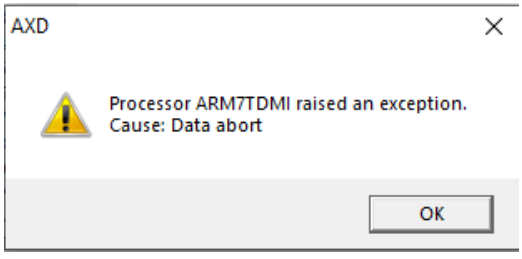
1. Εισαγωγή

Σκοπός της δεύτερης εργασίας είναι να δουλέψουμε στον τρόπο με τον οποίο ιεραρχείται η μνήμη και να παρουσιάσουμε κάποια αποτελέσματα και μετρήσεις. Θα ασχοληθούμε με τις περιοχές της μνήμης, πως χρησιμοποιείται η κάθε μια, καθώς επίσης με τις ταχύτητες τους και τι είδους δεδομένα αποθηκεύονται στην κάθε μνήμη. Επιπλέον, θα προσπαθήσουμε να βελτιώσουμε τη λειτουργία του συστήματος μας προσθέτοντας SRAM, για αποθήκευση δεδομένων που χρησιμοποιούνται περισσότερο. Τέλος, θα δοκιμάσουμε διάφορες μεθόδους βελτιστοποίησης των επαναλήψεων από την πρώτη εργασία για να δούμε πως αυτές διαφοροποιούνται ανάλογα με την εκάστοτε ιεραρχία μνήμης.

2. Ιεραρχία μνήμης χωρίς SRAM

Αρχικά, δηλώσαμε ότι χρησιμοποιούμε τις περιοχές της μνήμης ROM και RAM. Σε αυτές αποθηκεύονται τα read only data και όλα τα υπόλοιπα δεδομένα αντίστοιχα. Επιπλέον, οι θέσεις μνήμης που χρησιμοποιούνται είναι ίδιες με αυτές του εργαστηρίου και ίσες με 4 bytes η κάθε μία. Αυτές οι περιοχές ορίζονται στο αρχείο memory.map και όπως φαίνεται και παρακάτω, ξεκινάει από το τη θέση μνήμης 0x0 και έχει 80000 θέσεις για τη ROM ενώ για τη RAM ξεκινάει από τη θέση 0x80000 και έχει 8000000. Η ROM δηλώνουμε ότι μόνο διαβάζει δεδομένα σε αντίθεση με τη RAM η οποία και διαβάζει και γράφει, αυτό η πρώτη το κάνει με μεγαλύτερη ταχύτητα.

Στη συνέχεια, τροποποιήσαμε το αρχείο heap του εργαστηρίου, το οποίο περιλαμβάνει τις θέσεις μνήμης heap και stack. Αυτό το κάναμε επειδή αυτές οι περιοχές στη μνήμη επικαλύπτονταν από την περιοχή RAM και η εκτέλεση του προγράμματος δεν τερματιζόταν. Το error το οποίο παίρναμε και οι νέες περιοχές του stack και heap κατά την εκτέλεση παρουσιάζονται παρακάτω :



Error εκτέλεσης

```
/* works */
config.heap_base    = 0x08000000;
config.stack_base   = 0x08200000;
```

Νέες περιοχές stack, heap.

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

Το αρχείο memory.map

```
ROM 0x0 0x08000000
{
ROM 0x0 0x80000
{
*.o ( +RO )
}
RAM 0x80000 0x08000000
{
* ( ram )
* ( +ZI )
}

}
```

Το αρχείο scatter.

Τα παρακάτω είναι screenshots με τα αποτελέσματα εκτέλεσης αρχικά με τη μέθοδο decomposition και έπειτα με τη μέθοδο averaging.

- **Decomposition**

(Αποτελέσματα make)

```
==============================================================================
Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      3360         40          0    3775212       7596   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     17264        354          0    3775512      13628   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                17618 (  17.21kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      17618 (  17.21kB)
==============================================================================
```

(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	249230810	363539595	275569189	68313805	70969307	0	98694498	513546799	28229012

- **Averaging**

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      3360         40          0    3775212       7596   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     17264        354          0    3775512      13628   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                17618 (  17.21kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      17618 (  17.21kB)
==============================================================================
```

(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	255106937	368427846	281940519	67673767	70160001	0	96103062	515877349	28250180

3. Ιεραρχία μνήμης με SRAM

Θέλοντας να βελτιστοποιήσουμε περισσότερο τη λειτουργία του προγράμματος μας εισάγουμε τη μνήμη SRAM.
Για να γίνει αυτό πραγματοποιούμε τις εξής αλλαγές :

- ➔ Αρχικά, στο αρχείο map προσθετουμε τη νέα περιοχή με όνομα SRAM, η οποία έχει μέγεθος 8000 θέσεων και ξεκινάει από τη θέση 0x08080000 όπου τελειώνουν οι προηγούμενες μνήμες. Είναι και αυτή rw και έχει πολύ μεγαλύτερη ταχύτητα με σχέση με την RAM.

```
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
```

- ➔ Στη συνέχεια, κάνουμε την ίδια αλλαγή και στο scatter file μας όπου αρχικά δηλώνουμε τις θέσει μνήμης με βάση αυτές που δηλώθηκαν και στο αρχείο memory map. Επιπλέον, δηλώνουμε και ποια δεδομένα θα γράφονται και θα διαβάζονται στην SRAM τα οποία τα ονομάζουμε (sram).

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

}
```

- ➔ Τέλος, πραγματοποιούμε και την τελευταία αλλαγή στον κώδικα μας, όπου δηλώνουμε τις μεταβλητές οι οποίες χρησιμοποιούνται περισσότερο να αποθηκεύονται στην νέα περιοχή που δημιουργήσαμε για την SRAM. Με αυτόν τον τρόπο θα καταφέρουμε να ελαχιστοποιήσουμε τους κύκλους και τον χρόνο εκτέλεσης του προγράμματος γλυτώνοντας έτσι έξτρα κύκλους που θα είχαμε στην περίπτωση που αυτά τα δεδομένα αποθηκεύονταν στην RAM. Παρατηρήσαμε ότι αυτές οι μεταβλητές είναι οι current_r, current_g, και current_b οι οποίες όμως απαιτούν πολύ μεγάλο εύρος περιοχών με

αποτέλεσμα να πρέπει να αυξήσουμε πολύ το μέγεθος της μνήμης SRAM και κατ'επεκταση την δραματική αύξηση του κόστους. Στην προσπάθεια μας να ελαχιστοποιήσουμε το κόστος αυξάνοντας όμως την απόδοση βρήκαμε ότι ο πιο αποδοτικός τρόπος είναι η αποθήκευση των δεδομένων gray οπως φαίνεται και στο παρακάτω screenshot.

```
13  #pragma arm section zidata="ram"
14  int current_y[N][M];
15  int current_u[N/2][M/2];
16  int current_v[N/2][M/2];
17  int resize_u[N][M];
18  int resize_v[N][M];
19  int current_r[N][M];
20  int current_g[N][M];
21  int current_b[N][M];
22  #pragma arm section
23
24  #pragma arm section zidata="sram"
25  int gray;
26  #pragma arm section
```

*Στη μεταβλητή gray αποθηκεύονται οι τιμές των pixel που θα πάρουν στην έξοδο τους.

Τα παρακάτω είναι screenshots με τα αποτελέσματα εκτέλεσης αρχικά με τη μέθοδο decomposition και έπειτα με τη μέθοδο averaging.

- **Decomposition**

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      3360         40          0    3775212       7596   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     17264        354          0    3775512      13628   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                17618 (  17.21kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      17618 (  17.21kB)
==============================================================================
```

(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	249230837	363539631	275569220	68313808	70969309	0	94040886	508893223	28229012

- **Averaging**

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      3360         40          0    3775212       7596   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     17264        354          0    3775512      13628   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                17618 (  17.21kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      17618 (  17.21kB)
==============================================================================
```

(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	255106964	368427882	281940550	67673770	70160003	0	94360650	514134973	28250180

4. Βελτιστοποιήσεις κώδικα

α. Χωρις SRAM

- Loop Unrolling **(Decomposition)**

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      4528         40          0    3775212       7776   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     18432        354          0    3775512      13808   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                18786 (  18.35kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      18786 (  18.35kB)
==============================================================================
```

(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	131752372	196296564	144369618	40370442	37477224	0	80628150	302845434	14155849

- Loop Unrolling **(Averaging)**

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      5608         40          0    3775212       7928   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     19512        354          0    3775512      13960   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                19866 (  19.40kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      19866 (  19.40kB)
==============================================================================
```

(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	134232243	197357016	146855945	39743258	36680855	0	79543254	302823312	14158495

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

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      5548         40          0    3775212       7932   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     19452        354          0    3775512      13964   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                19806 (  19.34kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      19806 (  19.34kB)
==============================================================================
```

(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	131895152	196292054	144585218	40187567	37484227	0	80630130	302887142	14163952

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

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      5528         40          0    3775212       7932   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     19432        354          0    3775512      13964   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                19786 (  19.32kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      19786 (  19.32kB)
==============================================================================
```

(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	134665258	197533676	147289015	39632983	36578958	0	79545234	303046190	14166598

b. Με SRAM

- Loop Unrolling **(Decomposition)**

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      4528         40          0    3775212       7776   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     18432        354          0    3775512      13808   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                18786 (  18.35kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      18786 (  18.35kB)
==============================================================================
```

(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	131752399	196296600	144369649	40370445	37477226	0	78239802	300457122	14155849

- Loop Unrolling **(Averaging)**

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      5608         40          0    3775212       7928   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     19512        354          0    3775512      13960   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                19866 (  19.40kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      19866 (  19.40kB)
==============================================================================
```

(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	134232270	197357052	146855976	39743261	36680857	0	78672042	301952136	14158495

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

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      5548         40          0    3775212       7932   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     19452        354          0    3775512      13964   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                19806 (  19.34kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      19806 (  19.34kB)
==============================================================================
```

(Αποτελέσματα statistics)

Debugger Internals

Internal Variables | Statistics

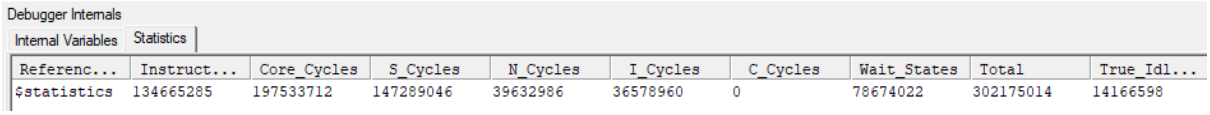
Referenc...	Instruct...	Core_Cycles	S_Cycles	N_Cycles	I_Cycles	C_Cycles	Wait_States	Total	True_Idl...
$statistics	131895179	195856490	144585249	39969770	37266429	0	78241782	300063230	14163952

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

(Αποτελέσματα make)

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

Image component sizes

      Code    RO Data    RW Data    ZI Data      Debug

      5528         40          0    3775212       7932   Object Totals
     13904        314          0        300       6032   Library Totals
==============================================================================

      Code    RO Data    RW Data    ZI Data      Debug

     19432        354          0    3775512      13964   Grand Totals
==============================================================================

    Total RO  Size(Code + RO Data)                19786 (  19.32kB)
    Total RW  Size(RW Data + ZI Data)           3775512 (3687.02kB)
    Total ROM Size(Code + RO Data + RW Data)      19786 (  19.32kB)
==============================================================================
```

(Αποτελέσματα statistics)



Referenc...	Instruct...	Core_Cycles	S_Cycles	N_Cycles	I_Cycles	C_Cycles	Wait_States	Total	True_Idl...
$statistics	134665285	197533712	147289046	39632986	36578960	0	78674022	302175014	14166598

5. Συμπεράσματα

Από τα τελικά αποτελέσματα βλέπουμε ότι ο κώδικας με την SRAM βελτιστοποιείται και μειώνονται οι κύκλοι. Γνωρίζουμε ότι η SRAM χαρακτηρίζεται από πολύ μεγαλύτερες ταχύτητες σε σχέση με τη RAM. Απο την άλλη, στην RAM υπάρχει μεγαλύτερο εύρος αποθήκευσης δεδομένων σε αντίθεση με την SRAM που έχει περιορισμένο. Αυτό αυξάνει αισθητά το κόστος της SRAM σε περίπτωση που αυτή χρησιμοποιείται σαν κύρια μορφή μνήμης.

Εν κατακλείδι, συμπεραίνουμε ότι πρέπει να βρίσκεται μια μέση λύση σε ότι αφορά τη χρήση των RAM σε συνδυασμό με τις SRAM μνήμες, έτσι ώστε να επιτυγχάνουμε το μέγιστο δυνατό αποτέλεσμα.