



Systemes Embarqués 1

Journal - TP.01 : Introduction

Auteurs :

M. Samuel RIEDO

samuel.riedo@edu.hefr.ch

M. Pascal ROULIN

pascal.roulin@edu.hefr.ch

Professeur :

M. Daniel GACHET

21 septembre 2016

Temps effectué hors des heures de classe

Nous avons travaillé jusqu'à 18 :40, donc 2 :05 de plus.

Synthèse des acquis

Non acquis :

- Language Assembleur

Acquis, mais à exercer encore :

- Utilisation d'Eclipse pour de l'Assembleur
- Utilisation de Git

Questions

Quelle est la taille de chacune des variables ?

- i : 4 bytes
- res : 4 bytes
- incr : 2 bytes

Quelle est la taille du code ?

Le code fait 0xc9b4 bytes en hexadécimal, ce qui correspond à 51636 bytes.

Comment procéder pour obtenir ces tailles ?

Les tailles des variables ont été obtenues en cherchant sur internet d'après leur type, la taille du code a été obtenue en lisant le champ ".text" dans le fichier "app_a.map".

Où se trouve chaque variable en mémoire (adresse absolue) ?

- i : 0x8000c900
- res : 0x8000df00
- incr : 0x8000df04

Où se trouve le code en mémoire ?

Le code commence à l'adresse mémoire 0x0000000080000000. Nous avons trouvé cette information dans le fichier "app_a.map". Nous n'avons par contre pas trouvé l'adresse de la fin du code.

Est-il possible d'améliorer l'algorithme ?

Oui, il est possible de l'améliorer de la façon suivante :

```

1  next:  ldr    r2, [r3]
2          add   r2, r1
3          str   r2, [r3]
4          ldr   r5, [r4]
5          add   r5, #1
6          str   r5, [r4]      // Update i eight times
7          cmp   r5, r0
8          bne   next
9
10  1:     nop
11          b     1b
12          bx    lr

```

L'opération associant la variable `i` à `r5` à chaque passage dans la boucle n'est pas obligatoire, nous avons donc modifié le code comme suit.

```
1  next:  ldr    r2, [r3]
2          add   r2, r1
3          str   r2, [r3]
4          add   r5, #1
5          cmp   r5, r0
6          bne   next
7          str   r5, [r4]          // Update i once
8
9  1:      nop
10         b     1b
11         bx    lr
```

Remarques

Faire attention aux push et pull lors de l'utilisation de Git afin de ne pas remplacer un fichier à jour en local par une ancienne version sur le serveur.

Feedback

Une introduction sur les différences entre la mémoire et les registres en début de cours, ainsi que les commandes leur étant associées, aurait été bienvenue en début de TP.