



Lab 06 – Arithmetics and addressing

R. Ferrero, M. Russo

Politecnico di Torino

Dipartimento di Automatica e Informatica (DAUIN)

Torino - Italy

This work is licensed under the Creative Commons (CC BY-SA) License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/3.0/>



Exercise 1

- Rename the registers:
 - r1 **to** single_value
 - r2 **to** double_value
 - r3 **to** triple_value
 - r4 **to** quadruple_value
 - r5 **to** quintuple_value
- Assign some value to single_value

Exercise 1 (cont.)

- By only using `MOV` and `ADD`, assign these values to the registers:
 - `double_value = single_value * 2`
 - `triple_value = single_value * 3`
 - `quadruple_value = single_value * 4`
 - `quintuple_value = single_value * 5`
- Suggestion: exploit Inline Barrel Shifter with `MOV`

Exercise 2

- Allocate 26 byte into a memory area DATA READWRITE, without initializing them
- Initialize $r0$ and $r1$ to 1
- Assign to registers $r2$ - $r12$ the elements of Fibonacci sequence (no loops). For example:
 - $r2 = r1 + r0$
 - $r3 = r2 + r1$
- Assign to $r14$ the address of the first byte of memory area allocated before

Exercise 2 (cont.)

- Using pre-indexed addressing, save the least significant byte of registers $r0-r12$, incrementing $r14$ at each assignment
- Using post-indexed addressing mode, save the least significant byte of registers $r12-r0$ (reverse order), incrementing $r14$ at each assignment.
- At the end, check that the content of the memory is the following:

01 01 02 03 05 08 0D 15 22 37 59 90 E9 E9 90
59 37 22 15 0D 08 05 03 02 01 01

Exercise 3

Write a program in ARM assembly language executing this tasks:

- Define the following constants in a code area
`myConstants DCW 57721, 56649, 15328, 60606, 51209, 8240, 24310, 42159`
- Allocate 16 byte (4 word) in a data area
- Considering the constants into couples, write in the 4 word the sum of the 4 couples of constants.