Lab #4
Bit-Wise Operations

Both parts of this exercise will count towards your final coursework mark for CS1021. Submit your solutions using Blackboard no later than 23:59pm on Monday 19th December 2016.

1 Shift-And-Add Multiplication by a Variable

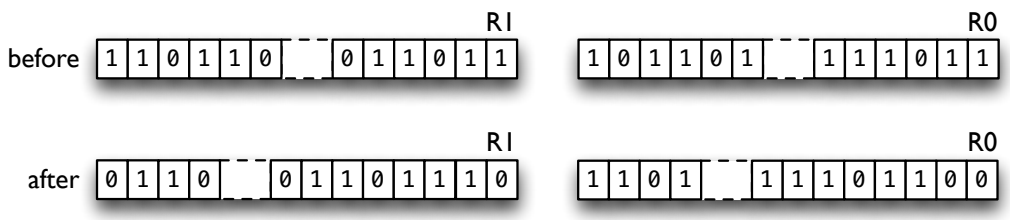
Design and write an ARM Assembly Language Program that will use shift-and-add multiplication to multiply the value in R1 by the value in R2, storing the result in R0. You may ignore any overflow problems.

Use the ShiftAdd project to develop your solution.

2 64-bit Shift

Design and write an ARM Assembly Language program that will perform a logical shift operation on a 64-bit value stored in registers `R0` and `R1`. Assume that the least-significant 32-bits are stored in `R0` and the most-significant 32-bits are stored in `R1`. The count of the number of bits to shift is stored in `R2`. If the count in `R2` is negative, your program should shift left. Conversely, if the count is positive, your program should shift right.

The following example shows how your program should logically shift left the 64-bit value stored in `R0` and `R1` when `R2` contains -2 (`0xFFFFFFFD` using the 2s Complement representation of negative values.)



Use the Shift64 project to develop your solution.

© UNIVERSITY OF DUBLIN 2016