Embedded Systems with ARM Cortex-M Microcontrollers in Assembly Language and C

Chapter 4
ARM Arithmetic and Logic Instructions
Exercises

Z. Gu

Fall 2025

Acknowledgement: Lecture slides based on Embedded Systems with ARM Cortex-M Microcontrollers in Assembly Language and C, University of Maine https://web.eece.maine.edu/~zhu/book/

Review

Barrel Shifter: Explanations

- LSL (logical shift left): shifts left, fills zeros on the right; C gets the last bit shifted out of bit 31. This is multiply by 2^n for non-overflowing values.
- LSR (logical shift right): shifts right, fills zeros on the left; C gets the last bit shifted out of bit 0. This is unsigned division by 2^n.
- ASR (arithmetic shift right): shifts right, fills the sign bit on the left to preserving the sign; C gets the last bit shifted out of bit 0. This is signed division by 2^n with sign extension
- ROR (rotate right): rotates bits right with wraparound; bits leaving bit 0 re-enter at bit 31, and C receives the bit that wrapped. This is a pure rotation without data loss.
- RRX (rotate right extended): rotates right by one through the carry flag, treating C as a 33rd bit; new bit 31 comes from old C, and C receives old bit 0.

Arithmetic with Shifts

- Assuimg 32-bit registers:
- Q1:
  - LDR r0, =0x00000007
  - MOV r0, r0, LSL 7
- Q2:
  - LDR r0, =0x00000400
  - MOV r0, r0, LSR 2
- Q3:
  - LDR r0, =0xFFFFC000
  - MOV r0, r0, LSR 2
- Q4:
  - LDR r0, =0xFFFFC000
  - MOV r0, r0, ASR 2
- Q5:
  - LDR r0, =0x00000007
  - MOV r0, r0, ROR 2

Assembly Programming

- Write ARMv7 assembly for pseudocode
  - r1 = (r0 >> 4) & 15