ARM Assembly

Lab 06

in_range
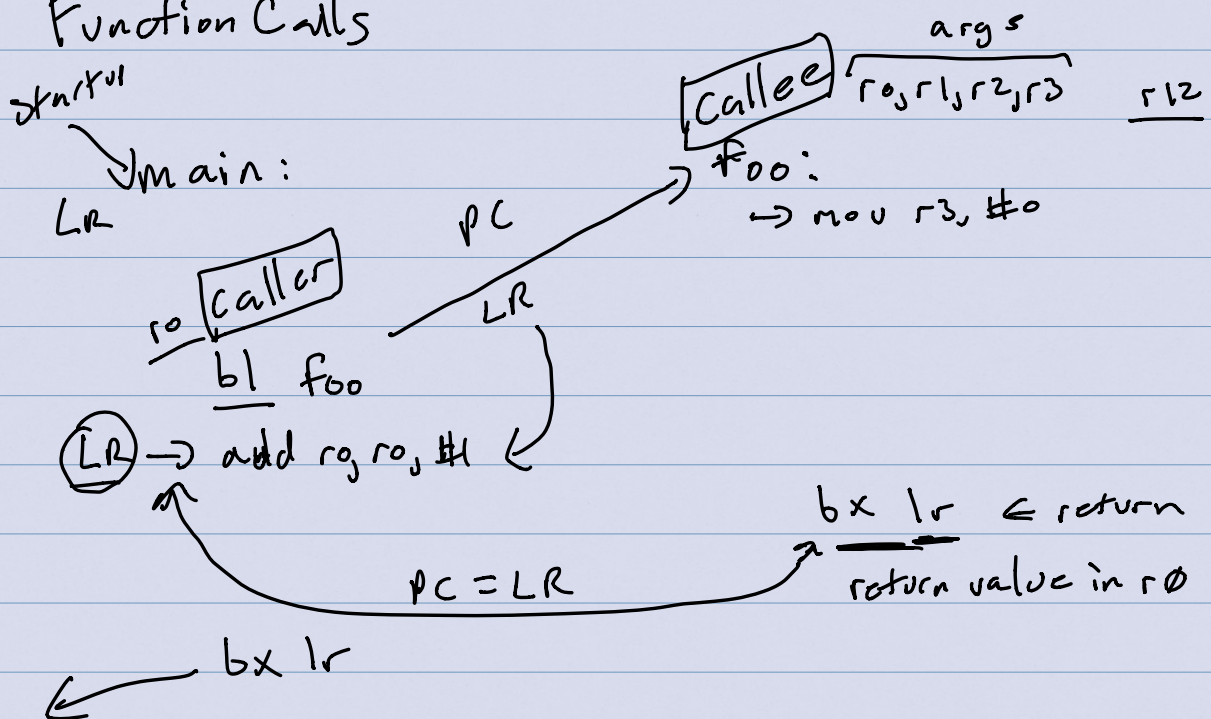
count_zeros

sum_array

Function Calls

add4_s using add2_s

Function Calls

~~initial~~

→ main:

LR

ro [Caller]

bl foo

(LR) → add r0, r0, #1

[Callee] | args : r0, r1, r2, r3   r12

foo:

→ mov r3, #0

PC

LR

PC = LR

bx lr ← return

return value in r0

bx lr

bl   branch and link

bx   branch and exchange

pc   program counter

lr   link register

Registers

caller – saved registers   callee saved registers

$r_0, r_1, r_2, r_3, r_{12}, lr$ $r_4 - r_{11}$ sp

r_0

|

r_{10}

r_{11} fp

r_{12} ip

r_{13} sp

r_{14} lr

r_{15} pc

foo:

　　sub sp, sp, #8

　　str r4, [sp]

　　⋮ r4

　　ldr r4, [sp]

　　add sp, sp, #8

　　bx lr

sp →

sp-8 →   r4

Proccessor

r0 r1 r2 r3

-- --.

r12 r13 r14 r15

STACK

foo:
    sub sp, sp, #8
    str lr, [sp]

    bl bar

    ldr lr, [sp]
    add sp, sp, #8
    bx

bar:

    bx lr