ARM Assembly
Lab 06
- in-range
- count-zeros
- sum-array
- Function calls
  - add4 —> using add2

Function calls

Caller main: Callee foo: args
                                                           $[r0, r1, r2, r3]$ $r12$

mov r0, #1
mov r1, #2   —> mov r3, #0
LR
↓ . bl Foo   PC
LR
LR —> add r0, r0, #1   bx $\boxed{lr}$ ← return

                                      return value in r0

addr
PC=LR

---
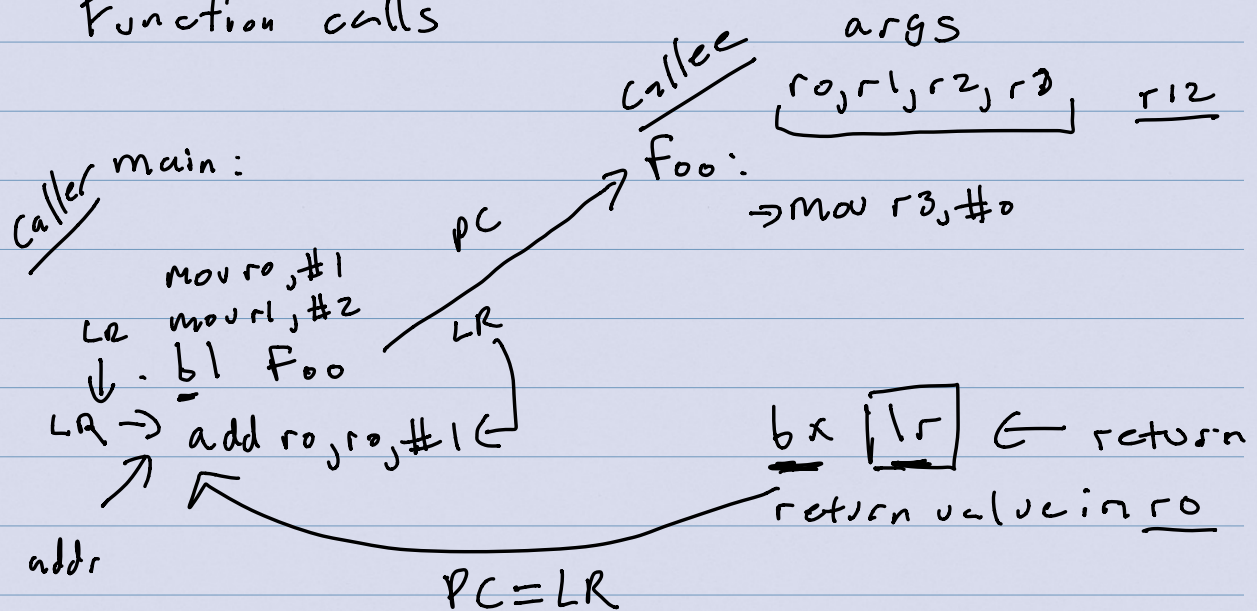
bl   branch and link
bx   branch and exchange
PC   program counter
lr = link register

Registers

caller-saved registers | callee-saved register

register values
that caller wants
to preserve

| | $r4 - r11, sp$

$\underline{r0}, \underline{r1}, \underline{r2}, \underline{r3}, \underline{r12, lr}$

Registers

$r0$

↓

r_{10}
$r11$ $\underline{fp}$
$r12$ ip
$r13$ $\underline{sp}$
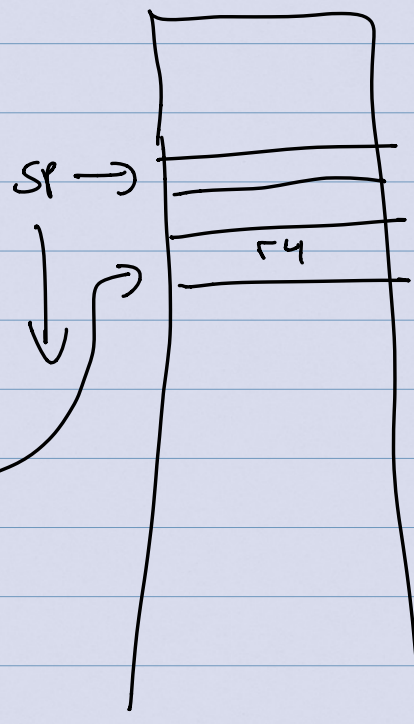$r14$ $\underline{lr}$
$r15$ $\underline{pc}$ *

foo:
   sub sp,sp,#8
   str r4,[sp]

   r4

   ldr r4,[sp]
   add sp,sp,#8

sp →

r4

bx lr