✓ gdb with ARM sorted out ← fib_rec_s, sum_array_sec_s, substr_s
✓ project 03 requirements
✓ project 03 questions
✓ Week08 Examples → Questions
✓ ARM 32 calling convations
→ ARM Instruction Set Format
ARM Reference
"Machine code"
Decoding
Encoding

---

gdb

compile and assemble with
-g

gdb progname

gdb command
"l" "list" list source code (C or Assembly)
l fib_rec_c
l fib_rec_s

"b" breakpoint

b function name

b label

b line number

"r" "run"

r <arguments>

When screen is messed up type

refresh -> CTRL-L

"n" next -> fine for C statements

not good for assembly instructions

"s" step "stepi"

"c" "continue" execute until

next breakpoint

project03

Makefile

$ make
=> project03
& make clean

project03.c ← driver

./project03 reverse "Hello World"

./project03 search 4 1 2 3 4 5

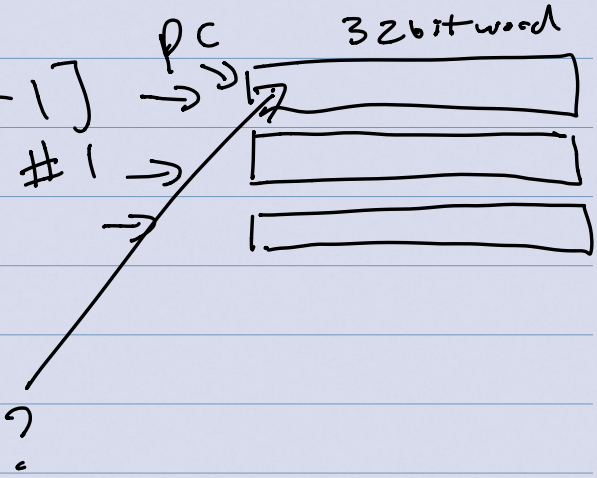
128 elements

./project03 sort 1 10 2 11 3

ARM Machine Code (RISC)

r0 = r0 + r1 + 1

```
add_s:
→ [add r0, r0, r1]
  add r0, r0, #1
  bx lr
```



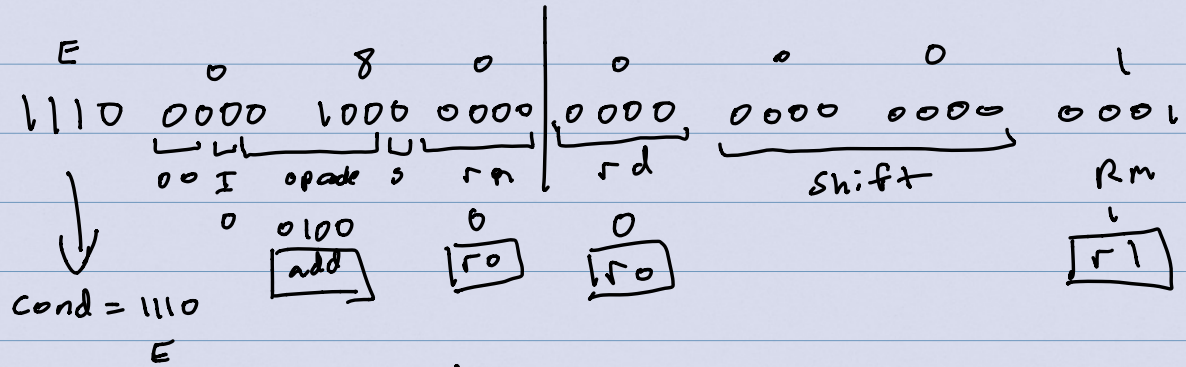
Data Processing

add, sub, cmp, mov

beq

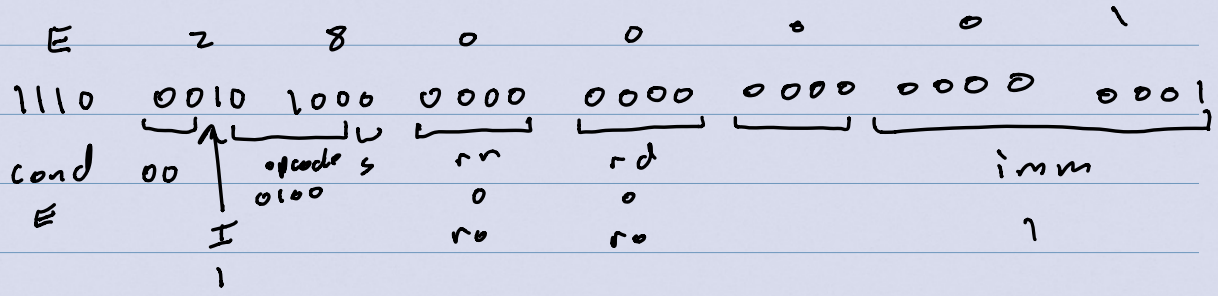
addeq

0xE0800001



add r0, r0, r1

0xE280 0001



add r0, r0, #1

bx lr

0xE12FFF1E