

C as implemented in Assembly – Tutorial Exercise

Study each of the following snippets of the disassembly of the C function `fn()` and answer the questions.

Snippet #1

```

3: void fn(int8_t *a, int32_t *b, float *c){
4:     volatile int8_t a1, a2;
5:     volatile int32_t b1, b2;
6:     volatile float c1, c2;
7:
0x08000370 B570      PUSH      {r4-r6,lr}
0x08000372 B086      SUB       sp,sp,#0x18
0x08000374 4604      MOV       r4,r0
0x08000376 460D      MOV       r5,r1
0x08000378 4616      MOV       r6,r2
...
```

Explain what each assembly instruction does and describe what data is in the register.

Snippet #2

```

8:     a1 = 15;
0x0800037A 200F      MOVVS    r0,#0x0F
0x0800037C 9005      STR       r0,[sp,#0x14]
9:     a2 = -14;
0x0800037E 20F2      MOVVS    r0,#0xF2
0x08000380 9004      STR       r0,[sp,#0x10]
10:    *a = a1 * a2;
11:
0x08000382 F89D0014  LDRB     r0,[sp,#0x14]
0x08000386 F89D1010  LDRB     r1,[sp,#0x10]
0x0800038A 4348      MULS     r0,r1,r0
0x0800038C B240      SXTB     r0,r0
0x0800038E 7020      STRB     r0,[r4,#0x00]
...
```

Explain what each assembly instruction does and describe what data is in the register.

Snippet #3

```

12:      b1 = 15;
0x08000390 200F      MOVS      r0,#0x0F
0x08000392 9003      STR       r0,[sp,#0x0C]
13:      b2 = -14;
0x08000394 F06F000D  MVN      r0,#0x0D
0x08000398 9002      STR       r0,[sp,#0x08]
14:      *b = b1 * b2;
15:
0x0800039A E9DD1002  LDRD     r1,r0,[sp,#0x08]
0x0800039E 4348      MULS      r0,r1,r0
0x080003A0 6028      STR       r0,[r5,#0x00]
...

```

Explain what each assembly instruction does and describe what data is in the register.
How does this compare with that in Snippet #2?

Snippet #4

```

16:      c1 = 15;
0x080003A2 4805      LDR       r0,[pc,#20] ; @0x080003B8
0x080003A4 9001      STR       r0,[sp,#0x04]
17:      c2 = -14;
0x080003A6 4805      LDR       r0,[pc,#20] ; @0x080003BC
0x080003A8 9000      STR       r0,[sp,#0x00]
18:      *c = c1 * c2;
0x080003AA E9DD1000  LDRD     r1,r0,[sp,#0]
0x080003AE F00F80F  BL.W     __aeabi_fmul (0x080003D0)
0x080003B2 6030      STR       r0,[r6,#0x00]
19:  }
20:
0x080003B4 B006      ADD       sp,sp,#0x18
0x080003B6 BD70      POP       {r4-r6,pc}
0x080003B8 0000      DCW      0x0000
0x080003BA 4170      DCW      0x4170
0x080003BC 0000      DCW      0x0000
0x080003BE C160      DCW      0xC160

```

Explain what each assembly instruction does and describe what data is in the register.
How does this compare with that in Snippets #3 and #4?

Reference: IEEE-754 Floating Point Converter
<https://www.h-schmidt.net/FloatConverter/IEEE754.html>

Appendix: Disassembled Main Function

```
21: int main(void){
22:     int8_t res1;
23:     int32_t res2;
24:     float res3;
25:
0x080003C0 B50E      PUSH      {r1-r3,lr}
26:     fn(&res1, &res2, &res3);
27:
0x080003C2 466A      MOV       r2,sp
0x080003C4 A901      ADD       r1,sp,#0x04
0x080003C6 A802      ADD       r0,sp,#0x08
0x080003C8 F7FFFD2 BL.W      fn (0x08000370)
28:     while (1);
0x080003CC BF00      NOP
0x080003CE E7FE      B         0x080003CE
```

- End of Tutorial Exercise -