Chapter 13 – Bit Manipulation Abstractions

testbit (boi = 14)

```
    ldr  r4,=0x12345678          // 0x12345678 - target

    ldr  r2,=(1<<14)             // r2:  0x00004000 - mask
    and  r2,r4,r2                // r2:  0x00004000 - apply mask
```

setbit (boi = 15)

```
    ldr  r4,=0x12345678          // r4:  0x12345678 - target

    ldr  r2,=(1<<15)             // r2:  0x00008000 - create mask
    orr  r2,r4,r2                // r2:  0x1234d678 - apply mask
```

clearbit (boi = 14)

```
    ldr  r4,=0x12345678          // r4:  0x12345678 - target

    ldr  r2,=~(1<<14)            // r2:  0xffffbfff - inverse mask

    and  r2,r4,r2                // r2:  0x12341678 - apply mask
```

togglebit (boi = 20)

```
    ldr  r4,=0x12345678          // r4:  0x12345678 - target

    ldr  r2,=(1<<20)             // r2:  0x00100000 - create mask
    eor  r2,r4,r2                // r2:  0x12245678 - apply mask
```

getbit (boi = 28)

```
    ldr  r4,=0x12345678          // r4:  0x12345678 - target

    ldr  r2,=(1<<28)             // r2:  0x10000000 - create mask
    and  r2,r4,r2                // r2:  0x10000000 - apply mask
    lsr  r2,r2,#28               // r2:  0x00000001 - normalize bit of interest
```

putbit (boi = 22)

```
    ldr  r4,=0x12345678          // r4:  0x12345678 - target
    mov  r5,#5                   // r5:  0x00000005 - new bit of interest value

    ldr  r3,=~(1<<22)            // r3:  0xffbfffff - create inverse mask
    and  r2,r4,r3                // r2:  0x12345678 - apply mask
    and  r5,r5,#1                // r5:  0x00000001 - mask new bit
    lsl  r5,r5,#22               // r5:  0x00400000 - de-normalize
    orr  r2,r2,r5                // r2:  0x12745678 - apply mask
```

getbits (bois = 12-15)

```
   ldr  r4,=0x12345678              // r4:  0x12345678 - target

   ldr  r3,=(~(~0<<4))<<12          // r3:  0x0000f000 - create mask
   and  r2,r4,r3                    // r2:  0x00005000 - apply mask
   lsr  r2,r2,#12                   // r2:  0x00000005 - normalize bits of interest
```

putbits (bois = 16-19)

```
   ldr  r4,=0x12345678              // r4:  0x12345678 - target
   mov  r5,0xa                      // r5:  0xa        - new bits of interest value

   ldr  r2,=(~(~0<<4))              // r2:  0x0000000f - normalized mask
   and   r5,r2,r5                   // r5:  0x0000000a - mask new value
   lsl  r5,r5,#16                   // r5:  0x000a0000 - de-normalize new value
   lsl  r2,r2,#16                   // r2:  0x000f0000 - de-normalize mask
   mvn  r2,r2                       // r2:  0xfff0ffff - invert mask
   and  r2,r4,r2                    // r2:  0x12305678 - clear bits of interest
   orr  r2,r2,r5                    // r2:  0x123a5678 - merge new bits of interest value into target
```

getbits (dynamic mask creation)

```
   ldr  r4,=0x12345678              // r4:  0x12345678 - target
   mov  r5,#4                       // r5:  size (variable = 4)
   mov  r6,#12                      // r6:  offset (variable = 12)

   mvn  r2,#0                       // r2:  ~0
   lsl  r2,r5                       // r2:  ~0<<size
   mvn  r2,r2                       // r2:  ~(~0<<size)
   lsl  r2,r6                       // r2:  (~(~0<<size)) << offset

   and  r2,r4,r2                    // r2:  0x000050000 - apply mask
   lsr  r2,r2,#12                   // r2:  0x000000005 - normalize
```