

```

1  #include <MKL25Z4.H>
2
3  __asm int my_sqrt(int x) {
4      MOVS r4, #0x0          ; a = 0
5      LDR r5, =0xFFFF        ; b = 2^16
6      LDR r6, =0xFFFFFFFF    ; c = -1
7
8      LOOP
9          MOVS r7, r6          ; c_old = c
10         ADDS r6, r4, r5       ; c = (a + b)
11         LSRS r6, #0x1        ; c = c / 2
12         MOVS r2, r6          ; r2 = c
13         MULS r2, r2          ; r2 = c * c
14         SUBS r2, r0          ; r2 = c * c - x
15         BEQ ENDWHILE        ; if(c * c == x) goto ENDWHILE
16         BMI ELSEIF          ; if(c * c < x) goto ELSEIF
17         MOVS r5, r6          ; if(c * c > x) b = c
18
19     WHILE_COND
20         SUBS r1, r6, r7       ; c != c_old
21         BEQ ENDWHILE
22         B LOOP
23
24     ELSEIF
25         MOVS r4, r6          ; a = c
26         B WHILE_COND
27
28     ENDWHILE
29         MOVS r0, r6          ; r0 = c (return calculated square root)
30         BX lr
31 }
32
33 int my__sqrt(int x) {
34     int done = 0, a = 0, b = 1<<16, c = -1, c_old;
35     do {
36         c_old = c;
37         c = (a + b) / 2;
38         if(c * c == x) done = 1;
39         else if(c * c < x) a = c;
40         else b = c;
41     } while(!done && c != c_old);
42     return c;
43 }
44
45 int main(void) {
46     volatile int r, j=0;
47     r = my_sqrt(0);           // should be 0
48     r = my_sqrt(25);          // should be 5
49     r = my_sqrt(133);         // should be 11
50     while (1);
51 }
52
53 // my_sqrt - 19 instructions
54 // my__sqrt - 30 instructions
55

```