Z80 Routines:Math:Multiplication

From WikiTI

Contents

Introduction

All these routines use the restoring multiplication algorithm, adapted to the z80 architecture to maximize speed. They can easily be unrolled to gain some speed.

Unsigned versions

8*8 multiplication

The following routine multiplies h by e and places the result in hl

```
mult_h_e
   ld   l, 0
   ld   d, l

   sla  h               ; optimised 1st iteration
   jr   nc, $+3
   ld   l, e

   ld b, 7
_loop:
   add  hl, hl
   jr   nc, $+3
   add  hl, de

   djnz _loop

   ret
```

16*8 multiplication

The following routine multiplies de by a and places the result in ahl (which means a is the most significant byte of the product, l the least significant and h the intermediate one...)

```
mult_a_de
   ld   c, 0
```

```
    ld   h, c
    ld   l, h

    add  a, a            ; optimised 1st iteration
    jr   nc, $+4
    ld   h,d
    ld   l,e

    ld b, 7
_loop:
    add  hl, hl
    rla
    jr   nc, $+4
    add  hl, de
    adc  a, c            ; yes this is actually adc a, 0 but since c is free we set it to zero and so we can save 1 byte an

    djnz _loop

    ret
```

16*16 multiplication

The following routine multiplies bc by de and places the result in dehl.

```
mult_de_bc
    ld   hl, 0

    sla  e               ; optimised 1st iteration
    rl   d
    jr   nc, $+4
    ld   h, b
    ld   l, c

    ld   a, 15
_loop:
    add  hl, hl
    rl   e
    rl   d
    jr   nc, $+6
    add  hl, bc
    jr   nc, $+3
    inc  de

    dec  a
    jr   nz, _loop

    ret
```

Signed versions

8*8 multiplication

16*8 multiplication

Retrieved from "http://wikiti.brandonw.net/index.php?title=Z80_Routines:Math:Multiplication&oldid=9864"
Categories: Z80 Routines:Math │ Z80 Routines

- ■ This page was last modified on 8 December 2011, at 17:10.