FastPlot

This plots a single over0 point on the screen, at speed.

It's been tested at a fill rate of over 8,500 points per second.

```
SUB fastPlot (x as uByte, y as uByte)
ASM
    ld d,(IX+5) ;'X
    ld e,(IX+7) ;'Y

    ld a, 191
    sub e
    ret c
    ld e, a
    and a
    rra
    scf
    rra
    and a
    rra
    xor e
    and 248
    xor e
    ld h, a
    ld a, d
    rlca
    rlca
    rlca
    xor e
    and 199
    xor e
    rlca
    rlca
    ld l, a
    ld a, d
    and 7
    ld b, a
    inc b
    ld a, 254

plotPoint_loop:
    rrca
    djnz plotPoint_loop
    ld b, 255
    xor b
    ld b, a
    ld a, (hl)
    or b
    ld (hl), a
END ASM
END SUB
```

Even faster, if you want to use the screen tables, is to lookup the screen address. `HRPrintFa` v: latest also uses the same table - it's important to only include it once - there's absolutely no benefit from including more than one copy. There, the magic is `include once` - and just have one copy of the source, to be sure it's the same one! You have to include this table (and the label "ScreenTables" somewhere in memory so the spectrum can find it.

This version has been tested with a fill rate of about 10,000 pixels per second (over 20% of the screen per second!).

```
SUB fastPlot (x as uByte, y as uByte)
ASM
    ld d,a ;'X
    ld a, 191
    sub (IX+7) ;' y
    jr c, plotPoint_end
    ld l,a
    ld h,ScreenTables/256
    ld a,(HL)
    inc h
    ld l,(HL)
    ld h,a
    ld a,d
    RRCA
    RRCA
    RRCA
    AND 31
    add a,l
    ld l,a

    ld a, d
    and 7
    ld b, a
    inc b
    ld a, 1

plotPoint_loop:
    rrca
    djnz plotPoint_loop
    ld b, a
    ld a, (hl)
    or b
    ld (hl), a
plotPoint_end:
END ASM
END SUB
```