HMirror.bas

This Function takes a byte in, and returns the byte that has reflected the bits around the x axis, such that bits 7,6,5,4,3,2,1,0 become bits 0,1,2,3,4,5,6,7.

This can be useful if you elect to make all your graphics face in one direction - you can mirror the bytes (perhaps to a buffer or a UDG) before you print them. It's faster to store them facing both ways, but you can make quite a memory saving if you just choose one way.

```
Function fastcall hMirror (number as uByte) as uByte
asm
   ld c,a
; unrolled loop for speed. Still quite small - costs 10 bytes over the loop version, and saves over ha
; 25 bytes and 96 clock cycles

   RR C
   RLA
   RR C
   RLA
   RR C
   RLA
   RR C
   RLA
   RR C
   RLA
   RR C
   RLA
   RR C
   RLA
   RR C
   RLA
end asm
END FUNCTION
```

The above function is basically deprecated, but may be easier to understand than the following. This one below is faster, and smaller. You should use this one:

```
Function fastcall hMirror (number as uByte) as uByte
asm
;17 bytes and 66 clock cycles
Reverse:
    ld b,a       ;b=ABCDEFGH
    rrca         ;a=HABCDEFG
    rrca         ;a=GHABCDEF
    xor b
    and %10101010
    xor b        ;a=GBADCFEH
    ld b,a       ;b=GBADCFEH
    rrca         ;a=HGBADCFE
    rrca         ;a=EHGBADCF
    rrca         ;a=FEHGBADC
    rrca         ;a=CFEHGBAD
    xor b
    and %01100110
    xor b        ;a=GFEDCBAH
    rrca         ;a=HGFEDCBA
end asm
end function
```