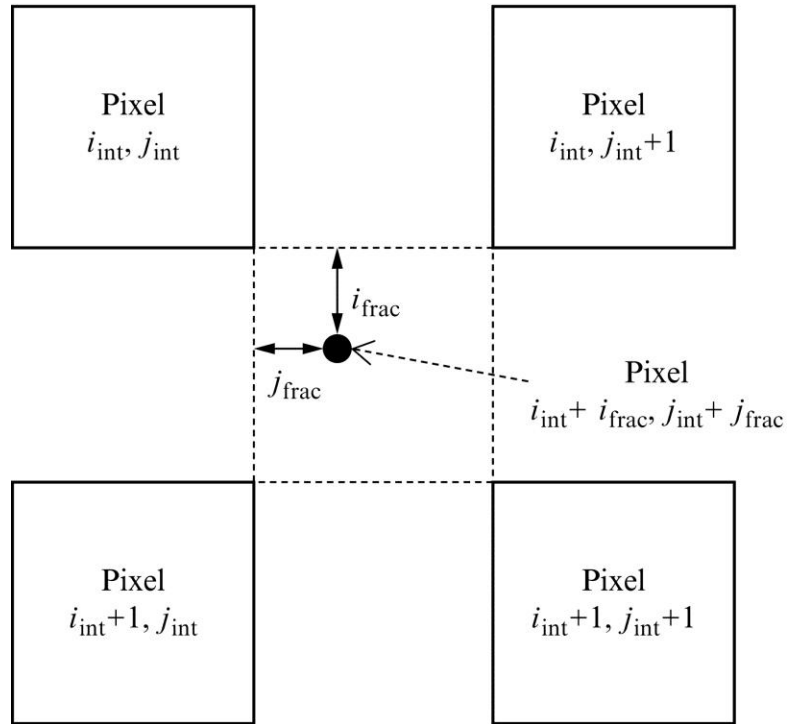
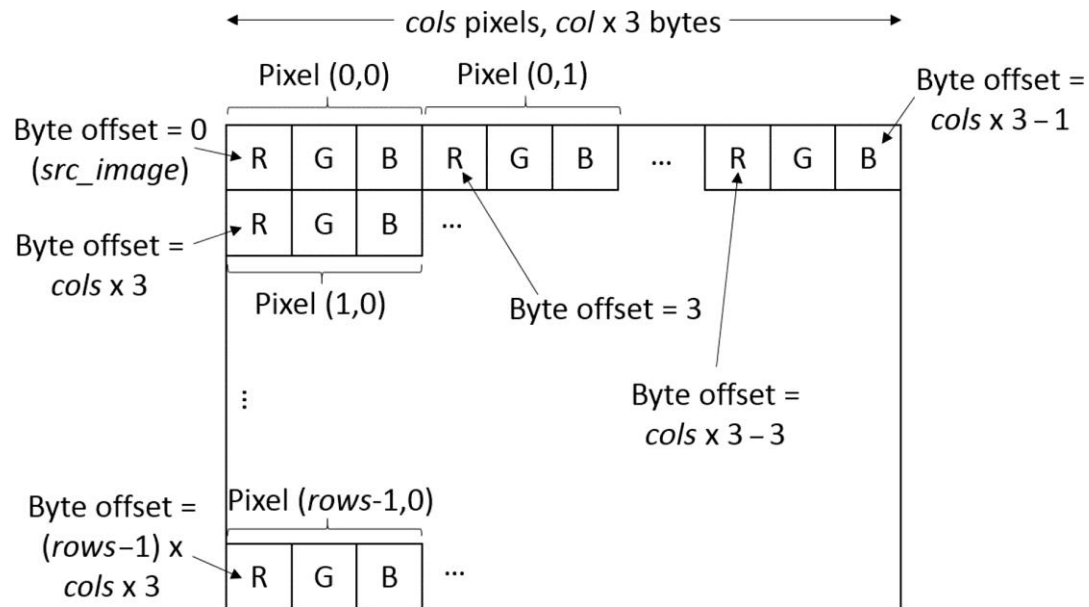


# Chapter 03

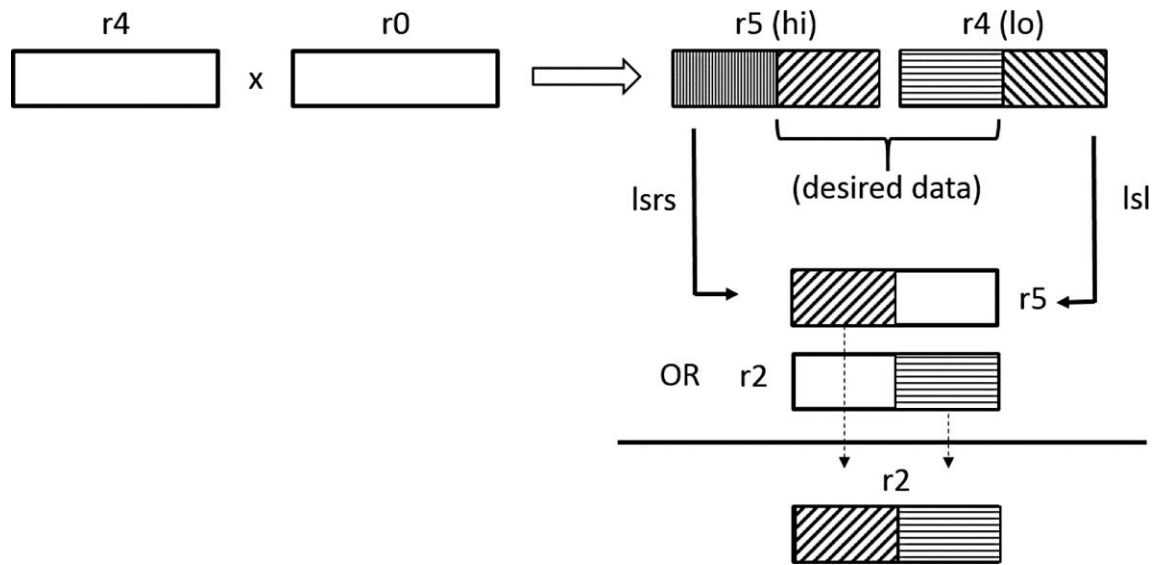
## Arithmetic optimization and the Linux Framebuffer



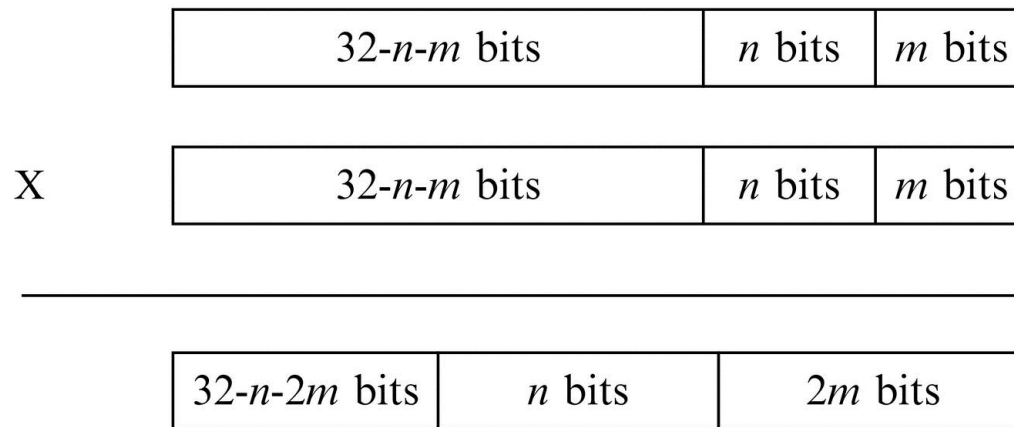
**FIGURE 3.1** Fractional source pixel.



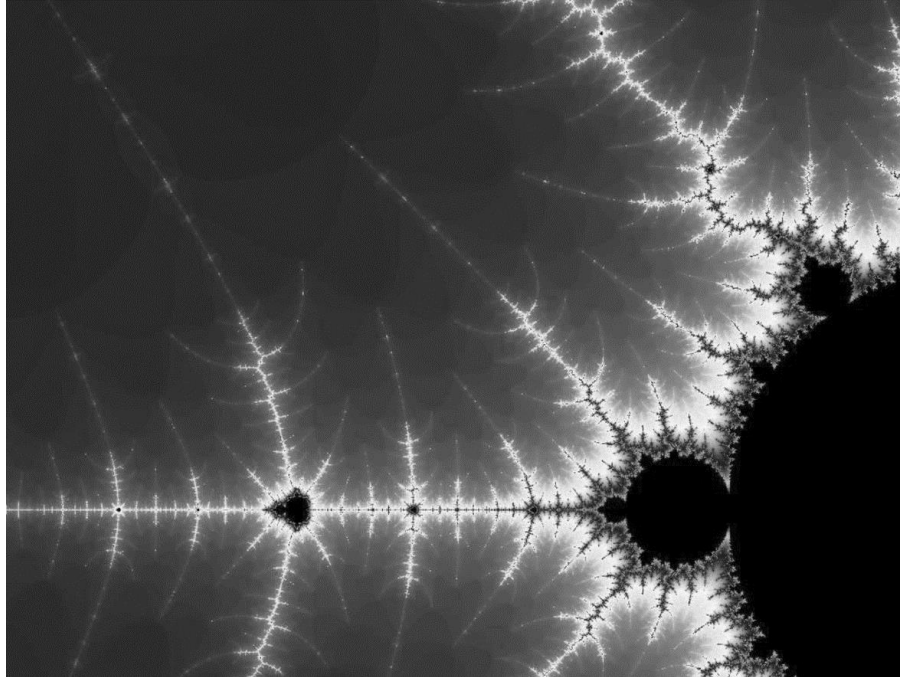
**FIGURE 3.2** Byte offsets for each color channel of each pixel within a 24-bit RGB frame.



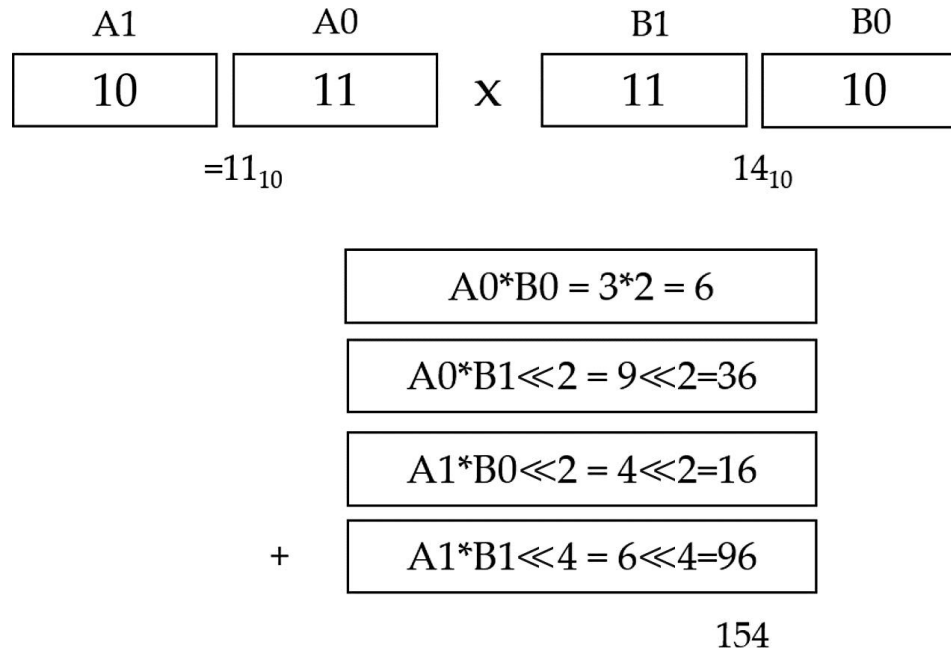
**FIGURE 3.3** Two (32,16) values multiplied to produce (64,32) product. To convert the product to a (32,16) value, the lower 16 bits of r5 and the upper 16 bits of r4 need to be combined into a single register. To do this, the compiler uses shift operations to isolate the bits and uses an OR instruction to combine them into a single register.



**FIGURE 3.4** Multiplying two  $(32, m)$  values when  $n+2m < 32$ .



**FIGURE 3.5** Mandelbrot set.



**FIGURE 3.6** An 8-bit multiplier implemented with 4-bit multipliers.