

Binary Numbers, Bits & Bytes

Let's do some counting!

0 1 2 3 4 5 6 7 8 9

Here we run out of digits, so we add a place and start again

10 11 12 13 ... and so on until we reach 99

where we also run out of digits, so we add another place, and restart

100 101 102 103 and so on

We have 3 places - the ones place, the tens place and the hundreds place. Mathematically these can be written as:

10^2 10^1 10^0 or again, 100's, 10's, 1's

Now, let's imagine that we only have the digits 0 and 1. Let's count:

0 start with 0
1 here we've run out of digits like we did with '9'
10 so we do this
11 and here we run out of digits again
100 so we do this
101 let's keep going ...
110
111
1000
1001
1010
1011
1100
1101
1110
1111 this in decimal is 15

The places here are powers of 2 rather than 10. We have a ones place, a twos place, a fours place, and an eights place. '1111' means 1 eight, 1 four, 1 two, and 1 one which is $8+4+2+1 = 15$

The places can be written as: 2^3 2^2 2^1 2^0

Let's add four more places: 2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0

The values of these are: 128 64 32 16 8 4 2 1

Each place in binary numbers is called a 'bit'. And 8 bits is called a 'byte'. With 8 bits we can count to 255 ($128+64+32+16+8+4+2+1$).

This should be a familiar number! How many values does each color have in PhotoShop? Each color, red, green and blue, has a value from 0 to 255. We often call this 24-bit color (8 bits of red, 8 green, and 8 blue).

So, why bits? Computers are made almost entirely out of switches, billions of them, and switches can have one of two values: off or on. Logically these can be true or false, or electrically, high or low.

First, let's build a picture. We have 1 byte each of red and blue in each of the smallest units of an image which is called a pixel. Let's imagine an 800 x 600 window. that's 480,000 pixels (or $\frac{1}{2}$ megapixel). Each pixel has 3 BYTES, which is $480,000 \times 3 = 1,440,000$ bytes. That's how much disk space this image will need! We usually call this 1.4 megabytes.

If we think about screens of video, we are changing 1.4 million bytes every 30th of a second. Since there are 8 bits in a byte and each switch is a bit, 1 second of video means we have addressed the values of $1,440,000 \times 8 \times 30 = 345,600,000$ nearly 350 million bits. That's a lot of data.

Let's think now about HD video which is 1920 X 1080 pixels. That's a huge amount of data: 1920×1080 pixels X 3 bytes per pixel X 8 bits per byte X 30 frames per second = $1,492,992,000$ bits or about 1.5 billion bits in a second.

A 'pict' file is the size in bytes based on the number of pixels as we figured above. A 'jpeg' is a file format that looks at regions of pixels and tries to save space by grouping similarities. "Make the sky blue" for example, uses fewer characters than writing all the numbers for every pixel. That said, jpeg compression does result in errors, called artifacts, because the sky isn't in fact a single color of blue. It's up to us to decide if we can tolerate the errors or if we need all the data. For that matter, digital color using 8 bits per color represents $256 \times 256 \times 256 = 16,777,216$ total colors which is an approximation of the infinite range of possible colors in the world. This isn't a problem since the consensus is that people can distinguish at best about 10 million different colors.

In video, the data streams are so huge that more complex schemes have been devised which look at both regions of pixels and also how pixels change in time, and use these similarities and repetitions to save data space. For example, we can say "show frame 4 twice". That saves some data space because it's simpler to say that than it is to actually write out the values of the two frames.

One additional note. Binary numbers are very useful for representing the values of switches used as sensors for a micro-controller such as an Arduino. For example, the number 11111111 can mean that 8 switches are high. 10101010 means every other switch is high. It's convenient to use this because we can use this single number, sent through a serial port, to control the information displayed in a multimedia program like Processing, PureData or Max, and is much simpler and more efficient than sending 8 separate numbers, one for each switch.