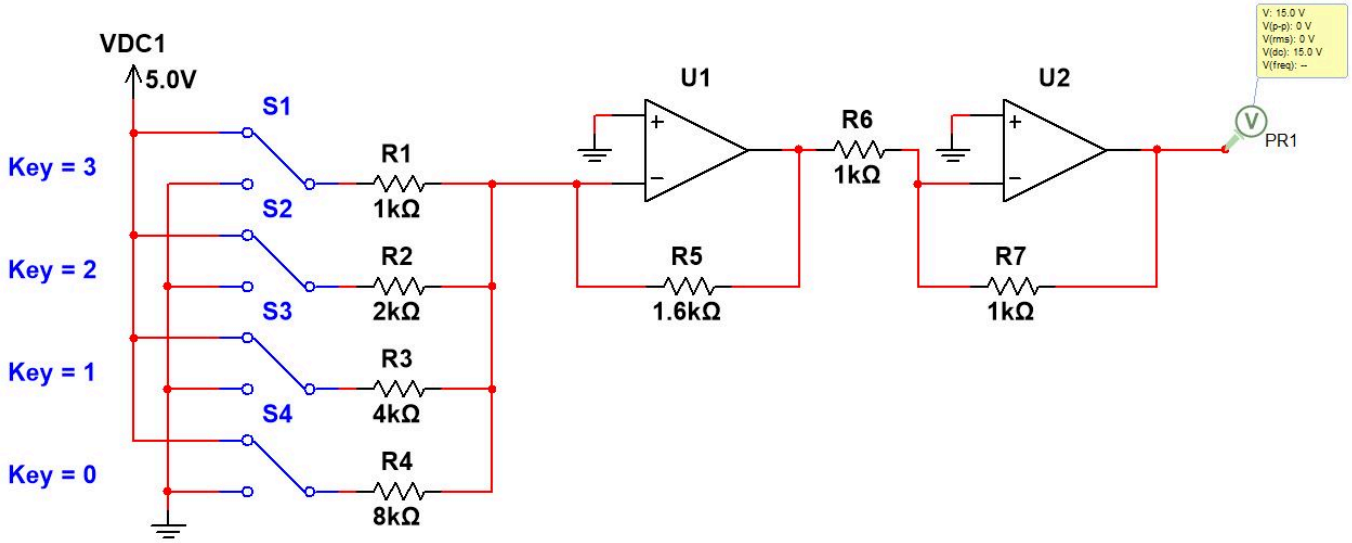
Digital to Analog Conversion, DtoA, is one half of the interface between the "real world" and binary devices. The other half is Analog to Digital Conversion (AtoD). AtoD Conversion is the process whereby an analog electrical signal generated by something in the real world is sampled and converted by an Analog to Digital Converter (ADC) into a binary representation and stored in a memory device. Digital to Analog Conversion is done by a Digital to Analog Converter (DAC) which takes binary values and generates the electrical signals they represent in order to produce something recognizable in the real world.

For example, consider an audio recording. A human voice generates constantly-changing sound pressure levels that are picked up by a microphone, which converts them, with the help of a signal conditioning circuit, into a constantly-changing voltage signal. An ADC samples the signal at a clearly-defined sample rate, generating a binary representation of each sample, which it sends to a memory device. To play back the recording, a DAC, running at the same sample rate, reads the stored binary values in the same order as they were generated and converts them into voltages which are sent by means of an amplifier to speakers or headphones to recreate sound pressure levels that are intended to be a reasonable representation of the original audio signal.

It seems like we should start with AtoD. However, DtoA is a simpler process. In fact, the most common ADC actually has, as just one of its components, a DAC. So, we'll start with DtoA and revisit AtoD after we've learned about voltage comparators.

The following circuit is a somewhat contrived 4-bit DAC with a huge step size -- not great for producing a good representation of an analog signal, but something that will help us learn how a DAC works.



Using Multisim, build this circuit.

The op amps are from ANALOG_VIRTUAL. You need to edit them so that the Positive voltage swing(VSW+) is +20 V and the Negative voltage swing(VSW-) is -20 V.

VDC1 is from Sources, and is V_REF1.

The switches are SPDT (single pole double throw) from SWITCH in Basic.

Run your circuit and fill in the following table, assuming that 0 V is logic LOW and +5 V is logic HIGH:

Binary input	Vout, V
0000	0
0001	1
0010	2
0011	3
0100	4

0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

Given that a Binary Step is from one number to the next, what is the step size for this DAC? 1 V/step

Mathematically, the step size can be determined a number of ways. If you know the entire range of values, you can use the following formula:

$$V_{step} = \frac{V_{range}}{2^n - 1}$$

...where n is the number of bits in the binary value.

This works because there will always be one less step than the number of voltage levels possible. Notice in our example that there are sixteen possible voltage levels -- 0 through 15, but if you start at zero, there are only 15 steps to the top. Check that on a staircase sometime! Also notice that you have five fingers on your hand, but only four gaps between them -- same concept.

Use this formula to determine the step size. 1 V/step

Electrically, for a DAC or ADC to work, they need a voltage reference. For reasons we'll see later, that voltage reference is always one step above the top end of the range. So, if you know the voltage reference, you can use the following formula:

$$V_{step} = \frac{V_{ref}}{2^n}$$

In our example, the reference voltage would be 16 VDC. Use this to determine the step size. 1 V/step

Looking back at the circuit, notice the strange resistor values -- only one or two of them are commercially-available. Imagine if this circuit had eight or sixteen input bits. The values would not only not be available, but the largest one would be 128x the size of the smallest one. Since this in an inverting amplifier, it means that the input impedances for each bit would be different. If the bit sources were non-ideal, this would introduce some pretty strange inaccuracies in the output voltage. We'll investigate a better DAC on the next page.