Digital to Analog and Analog-to-Digital Conversion

Laboratory Goal:	Build D-to-A and A-to-D circuits and demonstrate their operation
Learning Objectives:	A/D and D/A conversion principles, applications of op-amp circuits, mixed signal applications
Suggested Tools:	Oscilloscope, op-amp summing amplifier, op-amp comparator, microcontroller

The prelab is due at the beginning of the lab. For level 2, complete level 1 first because you will use the level 1 circuit in level 2.

Prelab

You pulled all the stuffing out of your younger sister's teddy bear and are installing a microcontroller, digital to analog converter (DAC), amplifier, and speaker inside it so you can program it to speak to her.

You decided to make a four bit DAC so it will sound retro robotic. (As opposed to a high bit converter that would sound more natural) You have a power supply module that can provide +15V and -15V. You know that the output pin of the TL081 op amps you are using can only accurately get within approximately 3 volts of the power supplies before they start to saturate. The microcontroller you are interfacing to your DAC has output pins that transition from 0V to 5V.

For the prelab assignment design a DAC similar to the one in figure 2 that will provide good dynamic range but not saturate the output of the DAC OP AMP.

Dynamic range can be defined as the ratio of the largest signal to the smallest signal that can be generated at the output of the converter. Since we know the maximum linear amplitude that the TL081 OP AMP can output is approximately three volts in magnitude less than the supply voltages, we can define the maximum output voltage of the converter to +-12 volts. For this type of DAC the maximum output is obtained by setting all the inputs to each corresponding bit to logic 'high' (5 volts) then setting the overall gain so that the output is approximately -12 volts. Note that maximum output is a magnitude measurement but its sign is negative in this case. For this circuit the minimum signal size is specified by the number of bits and it cannot be changed.

The lab, lab screencasts, and the Internet have good background material on summing OP AMP circuits and DACs if you find yourself confused about how to solve the prelab.

Level 1: Design, construct, and test a 4 bit binary weighted digital to analog converter.

Background

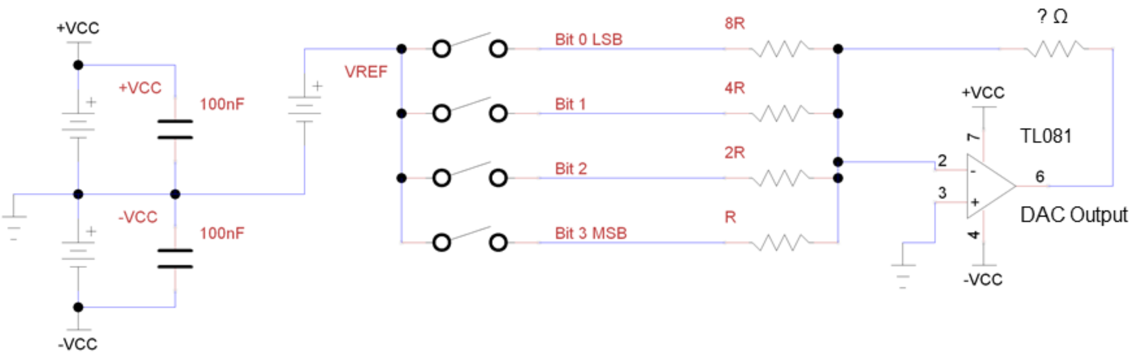


Figure 1: OP AMP summer four bit DA converter circuit. VCC+ and VCC- should be larger in magnitude than VREF. Switch Bit 0 is the least significant bit (LSB). Switch Bit 3 is the most significant bit (MSB). Pin 6 of the OP AMP is the output of the DA converter. The +- VCC power supplies only powers the OP AMP. The VREF only provides signal to the input of the summer. The value of the OP AMP feedback resistor should be chosen to maximize dynamic range of the OP AMP.

Digital signals typically have only two voltage states, a low and a high. Typically the low is specified as zero volts and the high is specified by the power supply voltage of the logic chips. To generate voltages that are not zero or power supply voltage a DAC is used. One of the simplest methods of converting from the digital to analog domain is by using an OP AMP summer circuit with appropriately weighted input resistors (figure 1).

The DAC functions by specifying different gains on the inputs of the summer. Because the same value logic voltage levels will be applied to each of the summer inputs, each input needs to have a different gain for it to result in a different output voltage. By selecting appropriate values for each of the summer input resistors the gains or weights of each input can be set. In figure 1 the values for the input resistors were set to give increasing binary weight to each input. The Bit 0 input is connected to the 8R (or 8 times R value) resistor which sets a gain of 1/8 (relative to Bit 3) for that input. This is the smallest gain therefore has the least significant impact on the output. The bit 3 input has an input resistor value of R therefore its gain is the largest. Since this input has the most significant effect on the output it is called the most significant bit (**MSB**). The feedback resistor on the summer in Figure 1 has a question mark for its value. You select the optimal value for this resistor so that when all the Bits are high the output will not saturate.

All four switches are connected to a voltage source called **VREF**. The value of VREF determines the magnitude of the output voltage from the DAC. Recall the summer is an inverting circuit. Therefore the sign of the output will be opposite of VREF. Also note that if VREF is too large in magnitude it is possible to saturate the output of the converter. This is likely to happen if all the inputs are in the on state and the resistor values set the overall gain too high.

Usually the input to the DA converter is provided by digital outputs (figure 2), not mechanical switches. A digital output can be thought of as a positive or negative going step function. The step transitions from zero to the VCC of the digital logic. The VCC of the digital logic may not be the same as that of the summer.

a) Design and construct a 4 bit binary weighted digital to analog (DA) converter.

b) Use a microcontroller for the binary inputs to the DAC as in figure 2. The microcontroller will replace the VREF and the four switches of figure 1 with four digital outputs. The VREF when using the microcontroller will be its supply Voltage (VCC). For the Arduino UNO the VCC of the microcontroller is 5 volts. If you use an Arduino to count through the DAC inputs, the EK307_DAC_LAB code on github.com/ekzosuls will provide the proper counting scheme. The code produces a repeating four bit binary count on Analog input pins A0 to A3. Note that the code reprograms the analog input pins to be digital outputs. The code produces a convenient periodic output to monitor on the oscilloscope. Setting the scope to trigger will make it easier to read the output.

c) If your converter is working properly its output will be a staircase with 16 voltage levels. Demonstrate the working circuit to your TA.

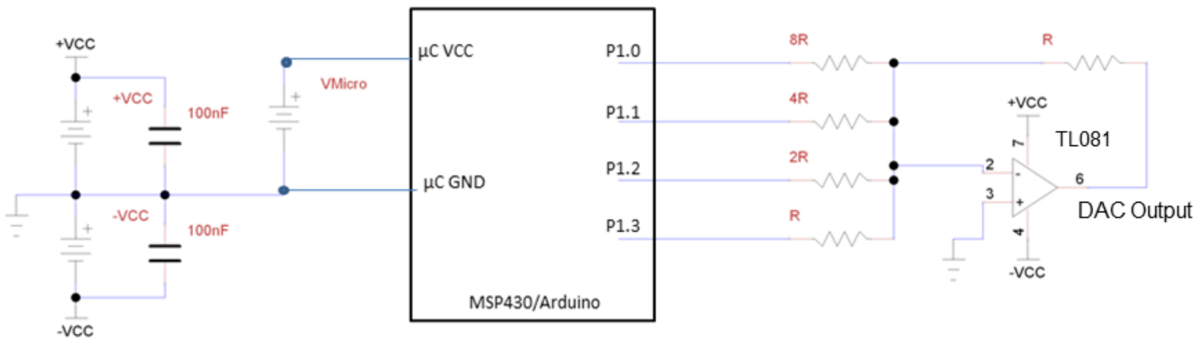
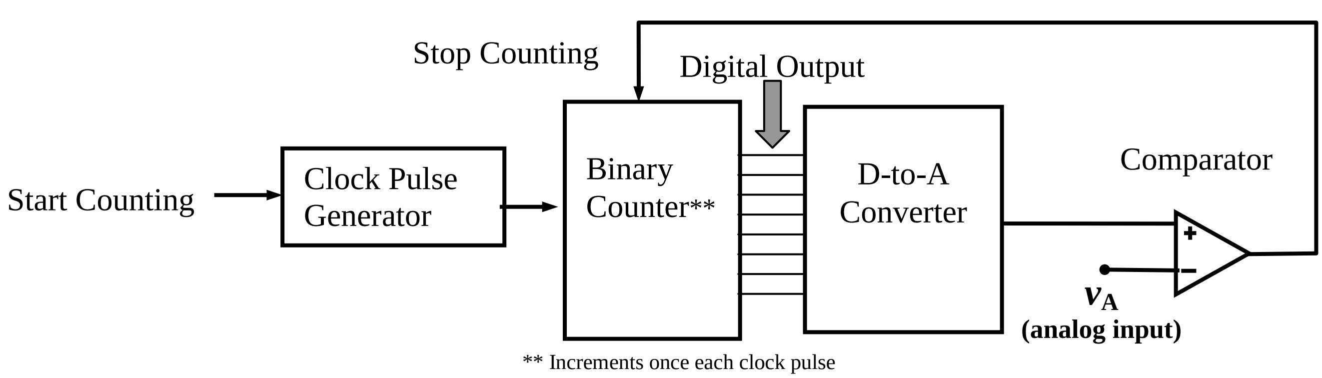


Figure 2: DA converter with a microcontroller (µC) providing the inputs. On many µC evaluation boards such as the Arduino Uno, the µC VCC power is provided over USB and you do not have to connect a power supply to the microcontroller. In this case it is important to know what the Voltage, VMicro, is so you can properly set the resistor values of the DAC to maximize dynamic range.

<u>Level 2 Binary-Weighted Analog-to-Digital Converter</u>

Design and demonstrate a 4-bit A-to-D converter.

One common method for designing an A-to-D converter (ADC) is to combine a D-to-A converter with a comparator and a counter that applies binary input digits to the D-to-A. The arrangement is shown below. The counter begins at binary 0000, then counts upward toward its highest value 1111. The upward counting increases the D-to-A output upward from zero. When the latter reaches the analog input v_A , the comparator "flips" state and freezes the counter. The counter output then constitutes the binary word representing v_A.



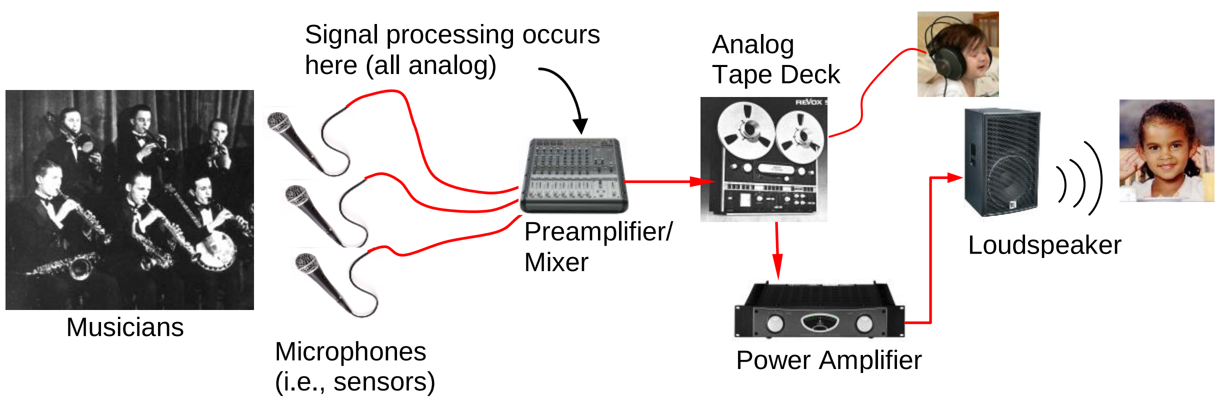
You can use a microcontroller as a counter to count up the DA converter. Feed the comparator output into the microcontroller. The microcontroller program will have to check the comparator state after each increment of the counter. Once the comparator output goes high then the count should stop and the binary value can be printed to the console. LEDs can be connected to the outputs of the binary counter to visually indicate the final value read in binary. Be sure to limit the LED current to less than 3mA to prevent loading of the microcontroller output pins. This is important because if the LEDs load the pins then the output of the microcontroller will not reach VCC and it will produce large error in the DA converter output Voltage.

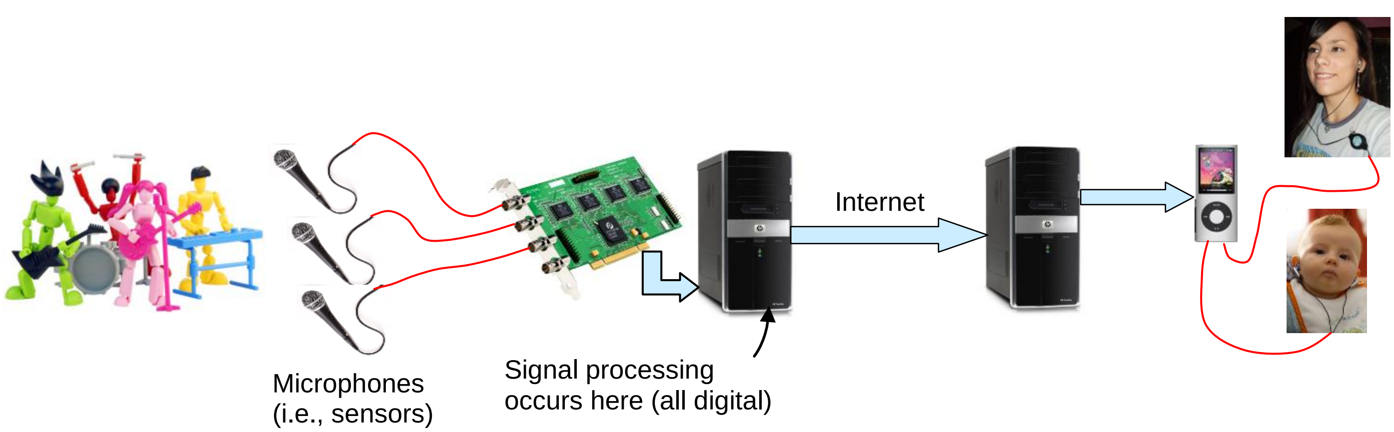
For this ADC scheme to work correctly the output of the DAC must go positive with increasing binary counts. The DAC in level one goes negative with increasing binary counts. How can you change or invert its' output to make the DAC suitable for level 2?

Appendix

Let's face it – the world we humans encounter is an analog playground, but nowadays, nearly all signal processing related to that world is performed digitally. Prior to the digital age, physical variables were measured in analog and were subsequently processed in analog too. Today, physical variables are still measured in analog form, and the resulting signals are immediately converted to digital form for processing, transmission, or storage on computers and digital devices. Compare, for example, the two diagrams shown below which illustrate how music was recorded in 1990 versus how it is recorded today. In each figure, analog signals are carried by *red* lines; digital signals by *blue* lines.

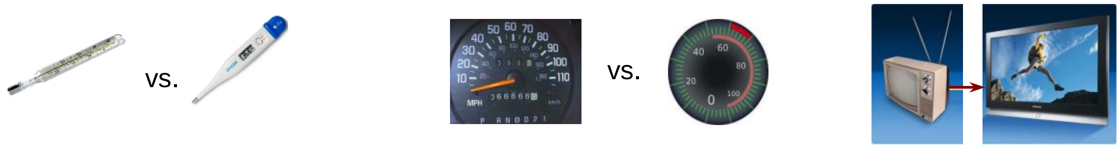
Recording and Listening to Music in 1990:



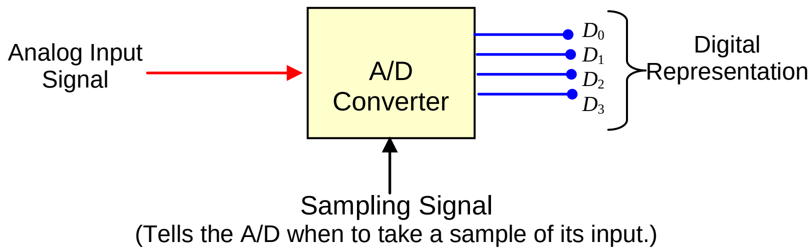


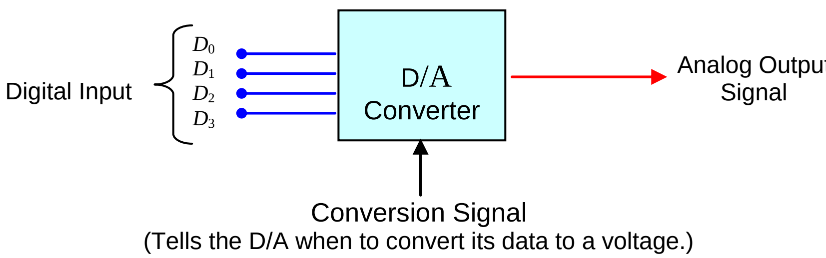
Recording and Listening to Music in 2015:

Most other sensing and signal processing jobs that used to be fully analog are now completely digital: the thermometer, speedometer, odometer, music, television, video, and movie projection.



Interfacing between nature's analog world and human-made digital systems requires a special class of circuits. Two important members of the interface family are the analog-to-digital (A/D) converter, and the digital-to-analog (D/A) converter. The diagrams shown below illustrate these conversion concepts in block form. In this laboratory assignment, you will build and otherwise become acquainted with each type of circuit.





Representing Analog Voltages in Digital Form

Let's assume that the physical variable serving as the object of our attention has been converted to a voltage by a sensor or transducer. (For example, a microphone converts sound waves into a varying voltage signal.) When an analog voltage v_A is converted to its equivalent digital representation, it's all done with respect to some user-defined reference voltage V_{REF}. The ratio v_A/V_{REF} determines the binary number value that will represent v_A. For this to happen, the voltage range between 0 and V_{REF} (or $-V_{REF}$ to V_{REF} in some systems) is divided into discrete voltage levels, where the latter are separated by the same small increment, called the converter's *voltage resolution*. The analog voltage v_A is assigned

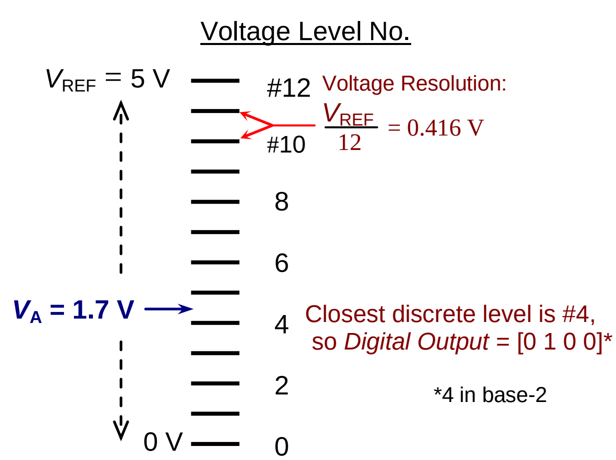


FIGURE 1

to the discrete voltage level lying closest to it in value. The concept is illustrated to the right for the very simplified case of a reference voltage of 5-V divided into 12 levels.

As noted above, the voltage increment separating the discrete levels is called the converter's *voltage resolution* (equal 0.416 V in the above example). The resolution determines the granularity, or fineness, with which v_A can be discerned by the converter.

In the above example demonstrates 12 discretization levels. In a more practical converter, the number will be at *least* 256 (for an 8-bit converter) and may be as high as 65,536 (for a state-of-the-art 16-bit converter.

For a more realistic example, lets look at an 8-bit converter with a reference voltage V_{REF} = 10 V. Eight bits of ones and zeros (1 and 0) yields 256 possible numbers from [0000 0000] to [1111 1111], i.e., from 0 to the base-10 number 255. The ratio v_A/V_{REF} determines which of the 255 levels, and hence which binary value, will be chosen to represent v_A. For example, if $v_A = V_{REF}/2$, the binary value of 128 is chosen (about half of 255). Similarly, if $v_A = V_{REF}/4$, a binary value of 64 is chosen (about one quarter of 255). If $v_A = V_{REF}$, then the assigned value would be 256, but that would require a ninth carry bit [1 0000 0000]. From this example, we see that an A/D converter can represent inputs *almost* up to V_{REF}, i.e., just one discretization level below.

The formulas pertinent to A/D and D/A conversion describes the relationship been analog voltages and their binary representations:

$$n = (2^N - 1)\frac{v_A}{V_{REF}} \qquad \text{and} \qquad v_A = \frac{n}{2^N - 1} V_{REF}$$

Here N is the maximum binary value possible (256 in the above example), and n is the decimal (base-10) equivalent of the binary representation of v_A. Here V_{REF} is, again, a reference voltage.

Analog-to-Digital Bin Converter

A *bin converter* is one in which the output consists of a simple linear row of digits, rather than a base-2 number. Its bits are set to **1** in linear sequence as v_A is increased. Bin converters are used, for example, in LED-based bar graphs. In the latter, the illuminated length of a row of lights indicates the relative magnitude of v_A. The bin converter determines the level to which the input should be assigned, as in Figure 1 above.

Bin Converter Concept

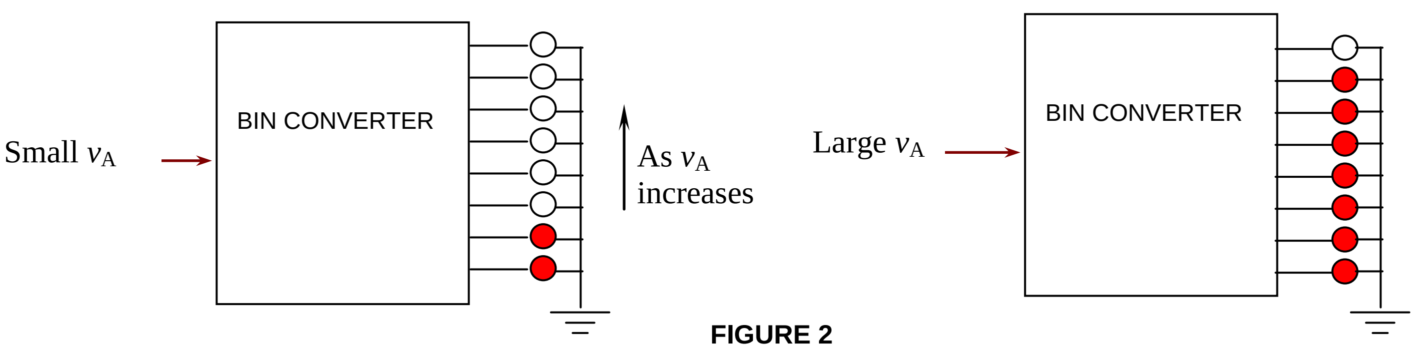


FIGURE 2

An op-amp connected as an open-loop *voltage comparator* is useful in designing a bin

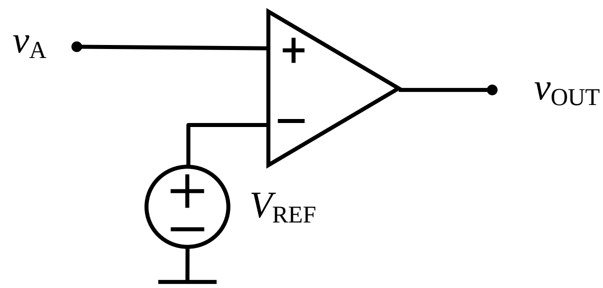


FIGURE 3

converter. In a comparator, the feedback path is absent. Recall that an op-amp will saturate with $\boldsymbol{v_{OUT} = V_{POS}}$ when $\boldsymbol{v_+ > v_-}$. Similarly, $\boldsymbol{v_{OUT} = V_{NEG}}$ if $\boldsymbol{v_+ < v_-}$. In the circuit shown to the right, the input v_A is compared to $V_{REF.}$ The op-amp output will be V_{POS} when $v_A > V_{REF}$. Conversely, the output will be V_{NEG} if
$v_A < V_{REF}$.