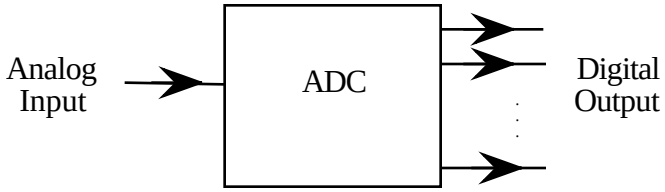
ECE 405 - SOURCE CODING - INVESTIGATION 17
INTRODUCTION TO CONVERTERS

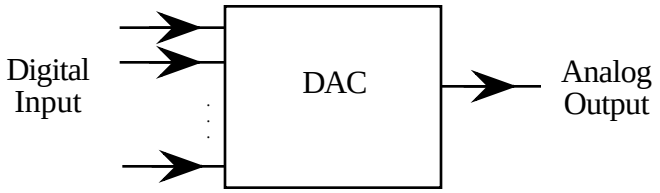
FALL 2005 **A.P. FELZER**

To do "well" on this investigation you must not only get the right answers but must also do neat, complete and concise writeups that make obvious what each problem is, how you're solving the problem and what your answer is. You also need to include drawings of all circuits as well as appropriate graphs and tables.

From the last Investigation we know how to calculate quantized PCM codes for samples m_s. The objective of this Investigation is to give an overview of how IC **Analog-to-Digital Converters (ADCs)** as follows



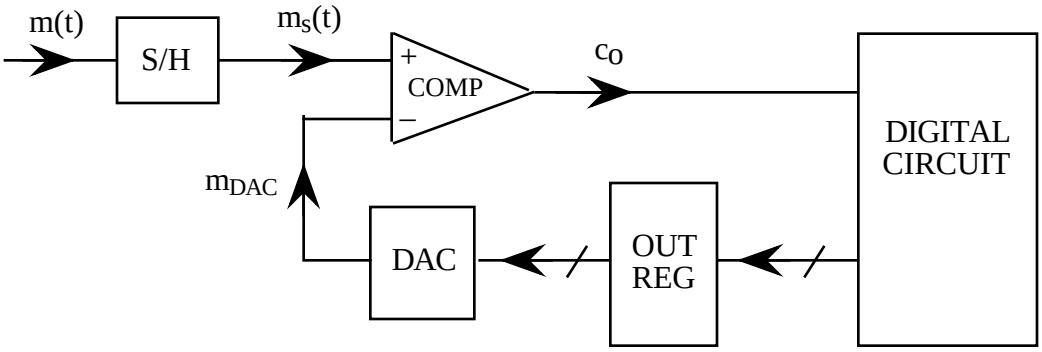
generate PCM code and how IC **Digital-to-Analog Converters (DACs)** as follows



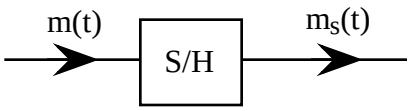
regenerate the analog sample values from the PCM code.

1. We begin with a review problem. Find the 8-bit signed binary PCM code for the following samples values of a message signal m(t) that varies from $-m_{max} = -10$ to $m_{max} = +10$
   a. $m_S = -2.3$
   b. $m_S = 8.7$

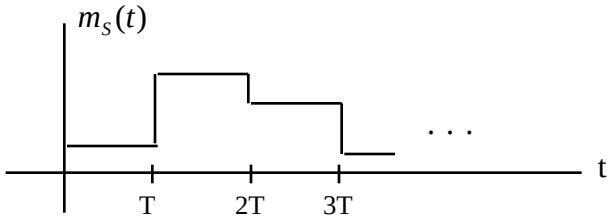
2. One of the most common types of ADCs is the **successive approximation** type as follows



where the sample-and-hold circuit



is a circuit that generates a staircase signal as follows



that "holds" sample values while they're being converted by the ADC.

Successive approximation ADCs can typically convert samples m_S to 12-bit PCM code at a rate of around 500 KSPS and to 16-bit PCM code at a rate of around 100 KSPS. Suppose in particular that the successive approximation ADC above is converting samples m_S ranging in value from 0 volts to 10 volts to 8-bit binary numbers. Note that we refer to 10 volts as the **Full Scale (FS)** value of the converter. Then the circuit will operate as follows for positive numbers

(0) The output register is first cleared to 0000 0000
(1) The digital circuit will then set bit 7 (the MSB) in the output register to 1. This makes the input to the DAC equal to 1000 0000 which will convert it to the corresponding analog signal $m_{DAC} = 0.5$ (FS) = 5 volts. Now

If $m_S \geq m_{DAC} = 0.5$ (FS) = 5 volts then $c_0 = H$ and bit 7 is kept set to 1
If $m_S < m_{DAC} = 0.5$ (FS) = 5 volts then $c_0 = L$ and bit 7 is reset to 0

(2) The digital circuit now sets bit 6 to 1. This makes the input to the DAC equal to

0100 0000 or 1100 0000

depending on the result of Step (1). Now

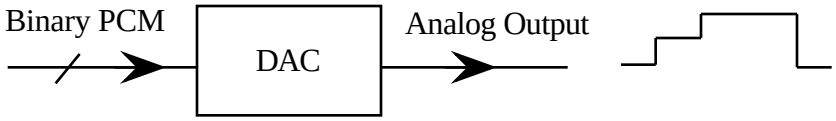
If $m_S \geq m_{DAC}$ then $c_0 = H$ and bit 6 is kept set to 1
If $m_S < m_{DAC}$ then $c_0 = L$ and bit 6 is reset to 0

(3) And so on for the rest of the bits

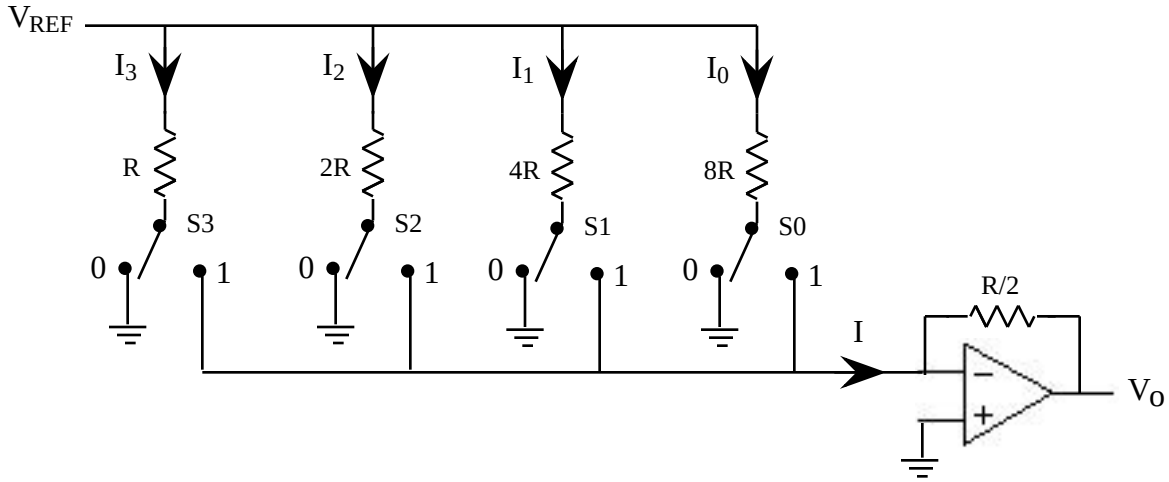
a. Write out Step 3 for the 3rd MSB of the successive approximation algorithm if the two MSB's are 01
b. Complete the following table for the conversion of $m_S = 6.3$ to 8-bit PCM code with FS = 10 volts

DAC Input	$\mathbf{m_{DAC}}$	$\mathbf{c_0}$
1000 0000		

3. The objective of this and the next problem is to introduce **digital-to-analog converters (DACs)** that generate "staircase" analog signals from their PCM codes as indicated in the following block diagram



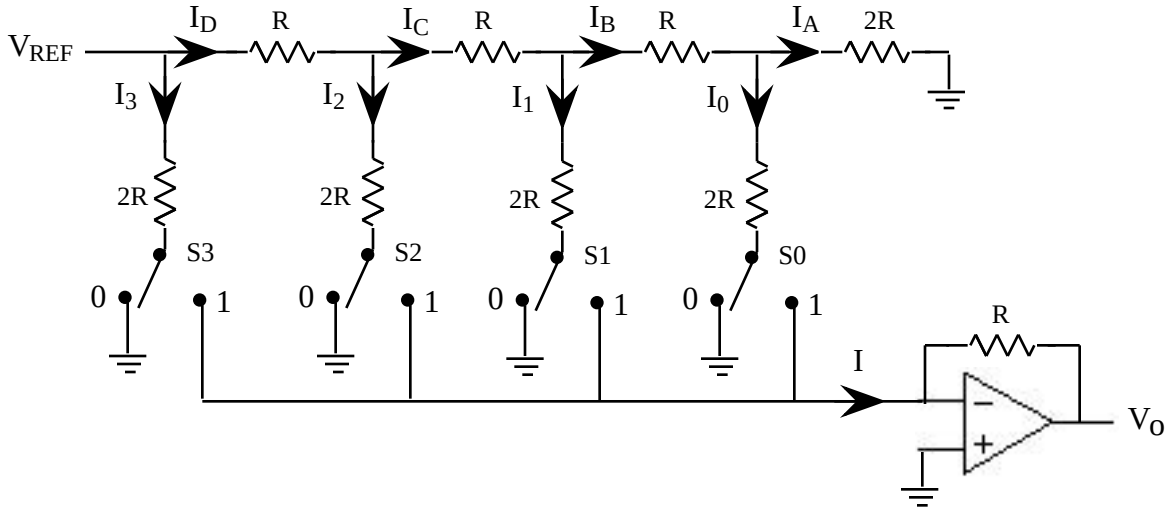
We begin with the following 4-bit DAC



with analog switches S_3, S_2, S_1 and S_0 controlled by the digits of the binary number $b_3b_2b_1b_0$ being converted

a. Why don't the currents I_3, I_2, I_1 and I_0 change when the switches change position
b. Find each of the currents I_3, I_2, I_1 and I_0
c. Express I_3, I_2 and I_1 in terms of I_0
d. Find I when the digital input is 0000
e. Find I when the digital input is 1111
f. Find I when the digital input is 1010
g. Make use of your results in parts (a)-(e) to explain how this DAC works

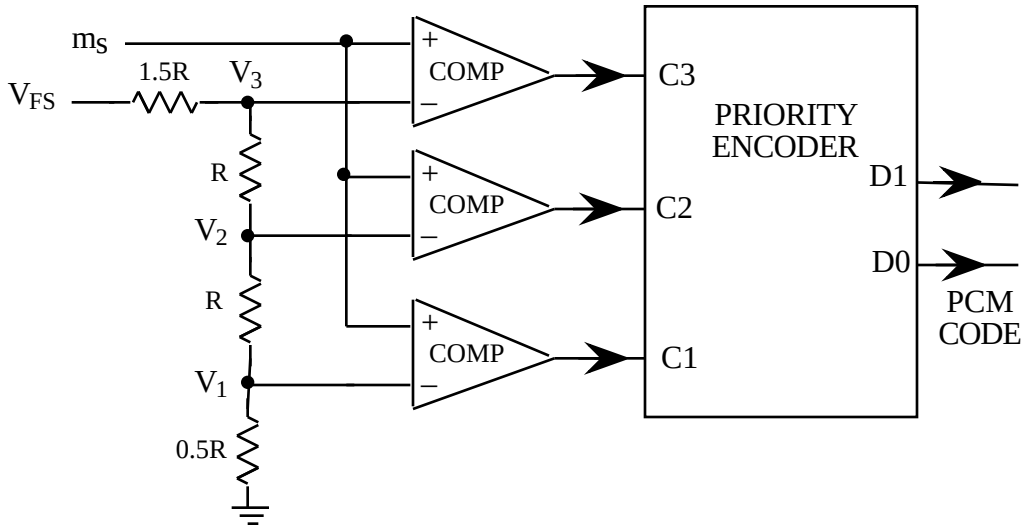
4. The DAC in Problem (3) is great except that for more than 4 bits the ratio between the resistor values becomes difficult to realize on an IC. An alternative design that doesn't have this problem is the following circuit with an R-2R ladder as follows



Verify that the currents I_3, I_2, I_1 and I_0 in this 4-bit DAC are 1/2 as much as those in Problem

(3) but generate the same V_o. Hint - find the equivalent resistance as seen by V_{REF}

5. Successive approximation ADCs like the one in Problem (2) are great but they take a "clock cycle" to generate each bit of the output. **Flash converters**, on the other hand, are harder to build because they generate an analog signal for every possible code but they're much faster. A simple 2-bit flash converter is as follows



Note that the output of a comparator is equal to 1 if $V_+ - V_- \geq 0$ and 0 if $V_+ - V_- < 0$.

a. Describe the operation of a priority encoder
b. Find V_3, V_2 and V_1 as a function of V_{FS}
c. Put values of V_3, V_2 and V_1 on the diagram when $V_{FS} = 10$ volts
d. Find the comparator outputs C_3, C_2 and C_1 and the converter output D_1D_0 if $V_{FS} = 10$ volts and $m_S = 1.3$ volts. Put your results in a Table
e. Repeat part (d) for $m_S = 6.3$ volts
f. Why is the flash converter faster than the successive approximation converter
g. How many comparators are needed for an 8-bit flash converter

6. The **dynamic range** of an ADC that is converting samples from $-m_{max}$ to $+m_{max}$ is given by

$$\text{Dynamic Range} = \frac{m_{max} - m_{min}}{\text{Resolution}} = \frac{2m_{max}}{\Delta} = 2^n$$

where n is the number of bits and the resolution is equal to the step size

a. Find the dynamic range in dB $(20\log_{10})$ of an ADC that converts signals from -10 volts to +10 volts with a resolution of 0.01 volts
b. Find the dynamic range in dB of a 12-bit ADC
c. Find the dynamic range in dB of an n-bit ADC
d. How many bits are needed for a dynamic range of 60 dB