EXPERIMENT 2

ANALOG TO DIGITAL CONVERTER (ADC)

Objectives

1. Learn how to design an ADC circuit that converts analog input signals to corresponding

binary output values.

2. Build, test, and troubleshoot an ADC circuit using Spice.

3. Hardwire the ADC circuit of objective 1 and compare the measurements of the hardwired

circuit with the measurements obtained using Spice.

Equipment Required

Components

1*ADC0804

1*10uF-POL

1*10Kohm resistance

1*10Kohm potentiometer

8*220ohm resistance

8*LED

Background Information

In many applications, ADCs and DACs are used together to provide a digital interface to an analog world.

Many ADC techniques are currently employed in industry; Parallel converter or Flash converter, Dual-Slope converter, Successive-Approximation converter are most widely used.

1. Parallel converter or Flash converter – uses many comparators connected in parallel, each with a different reference voltage. The outputs of these comparators are then input into a priority encoder. This encoder provides a binary output based on which comparator outputs are high and which are low. The number of bits the flash ADC accommodates determines the number of comparators used. e.g., an 8-bit flash ADC would require 28 – 1 = 255 comparators to implement the 8-bits. Therefore, while flash ADCs are the fastest converters, but they are limited in the number of bits they can accommodate. Generally, flash converters are limited to 4-bit up through 10-bit digital output resolutions.

2. Dual-Slope converter – this type of converter uses an op-amp integrator, comparator, digital control circuits, and digital clock circuits, along with a counter to generate the binary or BCD value from an analog source signal.

The fundamental idea is to switch the integrator input in a synchronized fashion between the analog signal to be converted and a known reference voltage (a negative voltage).This enables both charging and discharging of the integration capacitor at a constant rate. When the integration capacitor charges to a specific level, it triggers the

comparator. The comparator then triggers both discharging of the integrator capacitor and a counter that is used to count during the capacitor's discharge cycle. The value of the count is the amplitude of the analog input signal. This count value is then converted to a binary or BCD value and made available at the latch outputs of the ADC. Although the part count and circuit complexity for the dualslope converter is much less than that of the flash converter, the dual-slope converter is considerably slower due to the charge and discharge times required for its integrating capacitor.

3. Successive-Approximation converter – this type of converter employs the use of a DAC and a successive approximation register (SAR), along with control logic circuits, latch and output drivers to perform its ADC operations.Initially, all binary or BCD bits in the successive approximation register are set to zero. Once the conversion starts, the MSB is set to "1" and the bits (e.g., 10000000 for an 8-bit converter) are then fed, in a parallel fashion, into the DAC input, where the binary or BCD value is converted by the DAC into a reference voltage. This reference voltage is then fed into a comparator input where it is compared with the analog input signal. If the analog input signal is greater than the DAC derived reference voltage, the MSB is left as a "1" in the SAR; otherwise, it is set to "0". Then, this new binary value is shifted out to the DAC where a new reference voltage is generated and compared with the analog input signal. This function essentially provides a means of performing a binary search on the analog input voltage level.

This cycle continues until all binary bit values, from the MSB to the LSB, in the SAR have been set accordingly. Finally, the value is shifted out from the SAR into latch and output drivers to be made available outside of the ADC. Most of the off-the-shelf ADC Ics incorporate the SAR type converter.

Preliminary Work

Design an 8-bit ADC circuit that utilizes LEDs to indicate its binary output value. Use a reference voltage of 2.5V to 5 VDC. Design the circuit such that it will continually update its binary output for a changing analog input signal. Design the circuit to accommodate positive voltage (unipolar) operation only.

(Hint: Use two 1Kohm potantiometer, 1ADC, 2 kHz 5Volt square wave, 5volt DC and LEDs for output)

Simulate this Circuit with PSpice.

Experimental Procedure:

1. Build the circuit indicated in Figure 2 using a breadboard and actual components. Refer to the National ADC0804 data sheet for pin function descriptions.

2. Adjust R10 until Vin (pin 6) is equal to 0V.

3. Complete the following table of ADC binary output values for given voltage input values.

Conclusion

After completing the ADC circuit implementation in both PSPice and hardwiring, what are your conclusions? What similarities do the hardwired circuit and the circuit simulation share? What major differences did you notice between the two?

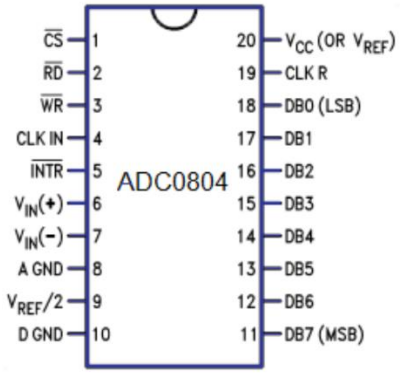


Figure 1. Pin configuration of ADC084

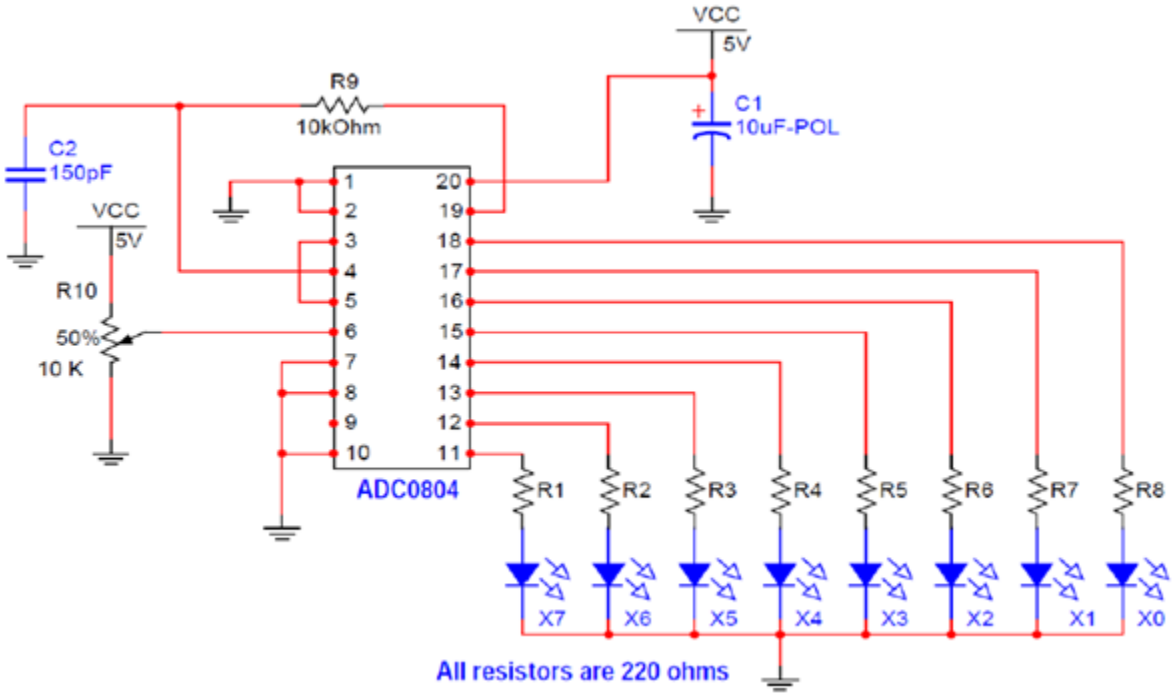


Figure 2. Circuit diagram for ADC experiment.

Vin (R10 setting)	Binary Output Calculated	Binary Output Indicated X7 → X0
0V		
0.020V		
0.040V		
0.080V		
0.160V		
0.320V		
0.640V		
1.280V		
2.500V		
3.750V		
4.375V		
4.680V		
5.000V		

Table 1