Engr 24L
Mission College

Lab 7
Analog to Digital Converter (ADC) – Part 1

This lab involves working with a simple ADC Integrated Circuit (IC). An Analog-to-Digital Converter (ADC, A/D, or A-to-D) is a device that converts a continuous physical quantity (usually voltage) into a digital number (digital codes or 1's and 0's). The binary codes represent the input's amplitude. In digital circuits, a '1' represents high voltage (usually 5V) and '0' represents a low voltage (usually 0V). In reality there is a range of possible voltages that are considered "high" and range of voltages that are considered "low".

ADCs are found in many applications as intermediate devices used to convert signals from analog to digital form. These digital signals are used for further processing by digital electronics. Various sensors like temperature, pressure, force etc. convert the physical characteristics into electrical signals that are analog in nature. The analog signals are then fed into an ADC circuit. What might be some benefits of converting an analog value into a digital number?

In this lab, we will use and test a conventional ADC IC., the ADC0804.

ADC0804 is a commonly used 8-bit analog to digital convertor. It is a single channel IC (Fig. 1). It can take only one analog signal input. The digital outputs vary from 0 to a maximum of 255. The step size can be adjusted by setting the reference voltage at Pin-9. When this pin is not connected, the default reference voltage is the operating voltage (V_{cc}). The step size at 5V is 19.5 mV. (5volts/256) Therefore, for every 19.5 mV rise in the analog input, the output increases by 1 unit (1 step). To set a particular voltage level as the reference value, pin-9 is connected to half the max allowed input voltage. For example, to set a reference of 4V (V_{ref}), Pin9 is connected to 2V ($V_{ref}/2$), thereby reducing the step size to 15.6 mV. Verify this step size by making a calculation. What would the step size be if the input were to only vary from 0 to 3V but you wanted a full range of digital values on the output?

The ADC0804 chip needs a clock to operate. The time to convert the analog value to a digital value depends on this clock source. An external clock can be applied to the Clock IN pin (pin-4). However the ADC0804 also has a built in clock that can be used in absence of an external clock. A suitable RC circuit is connected between the Clock IN and Clock R pins to use the internal clock. **The time constant of the RC circuit determines the clock speed., and therefore determines how fast the ADC chip samples the analog input on pin-6.**

Skim through the ADC0804 data sheet and see how much you can understand:
http://www.ti.com/lit/ds/symlink/adc0803-n.pdf

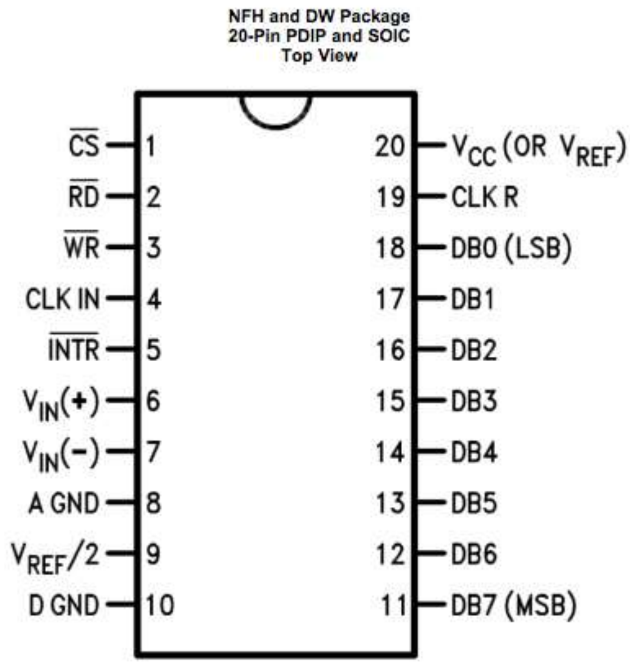


Fig. 1. Schematic of ADC0804. (From Texas Instruments®)

PART 1:

Equipment:

1) Oscilloscope
2) Power Supply – one signal for Vcc and another for the analog input
3) LRC meter to measure capacitance
4) Multimeter

Parts List for Lab:

1) Breadboard and connecting wires
2) ADC0804
3) 8 LEDs.
4) 150pF capacitor
5) 10 KΩ resistor
6) 2 similar, high-value resistor for making V_{ref} /2.
7) 8 pull up resistors for testing the circuit with LEDs. (1K Ω)
8) 8 pull up resistors for testing the circuit with logic analyzer. (2K Ω)

The schematic of the final system is shown in Fig. 2. Also, the picture of the real implementation is shown in Fig. 3.

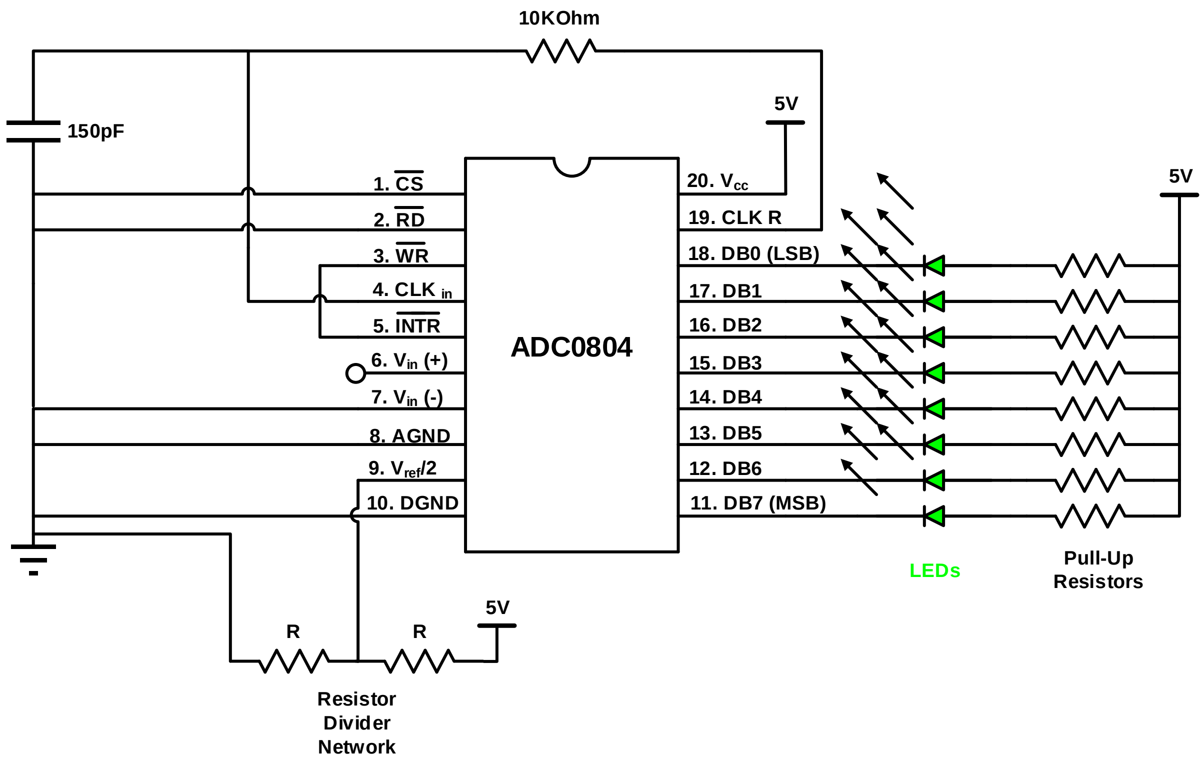
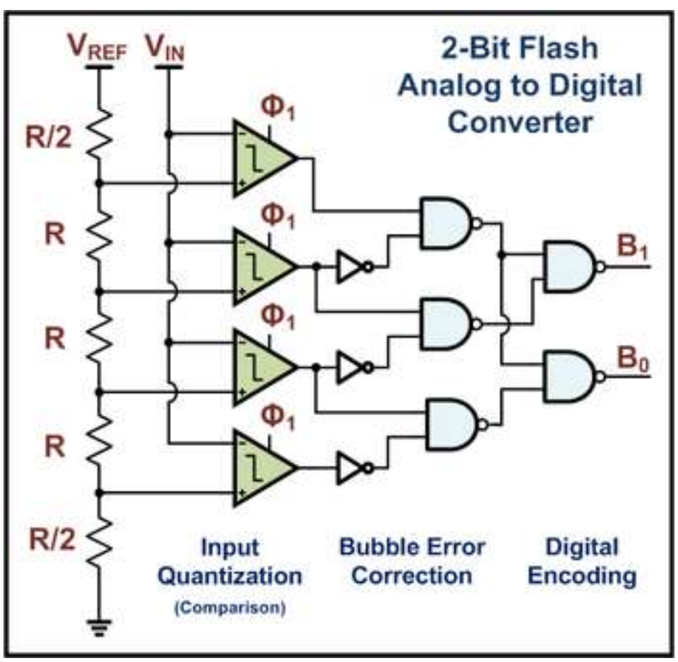


Fig. 2. System schematic.

Example ADC



Note that the LEDs are placed with the cathodes (negative side) into the pins 11-18. What might be the disadvantage of placing the LEDs with the anodes connected to pins 11-18?

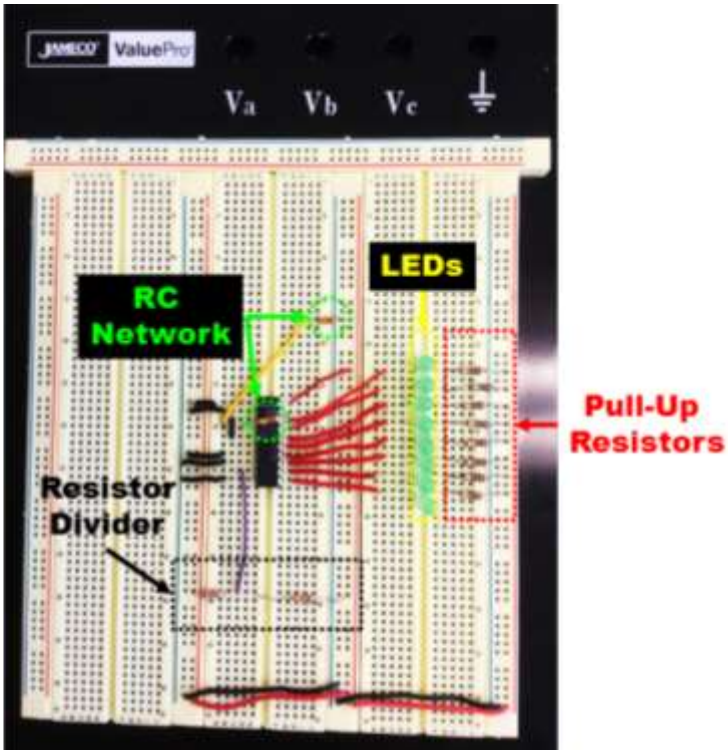


Fig. 3. Photograph of the implemented system on the breadboard.

Use the <u>large breadboard</u> for this lab. Use these steps when constructing the ADC system:

I) Place your IC on your breadboard.
II) Connect Pin 1 and Pin 2 to the ground.
III) Wire Pin 3 to Pin 5.
IV) Setup the RC network between Pin 4 and Pin 19.
V) Ground Pins 7, 8, 10.
VI) Build the voltage divider circuit and connect V_{ref} /2 (Pin 9) to the center point.
VII) Connect Pin 20 to V_{cc}.
VIII) Place 8 LEDs on your breadboard with appropriate space between them. See picture in Fig. 3.
IX) Wire each LED to a pull up resistor and then to the power supply.
X) Connect each of the LEDs to Pins number 11 to 18
XI) Set your voltages (Vcc = 5V and Vin = 0V) and set the OVP limit to 5V for both Vcc and Vin. Make sure OVP is turned "on". This way you won't accidentally put an input voltage over 5V and harm the chip. Set the current limiting to 200mA.

Problems you may encounter:

1) Place the long LED wire to the positive side (Vcc). The short wire should go to the chip pins.

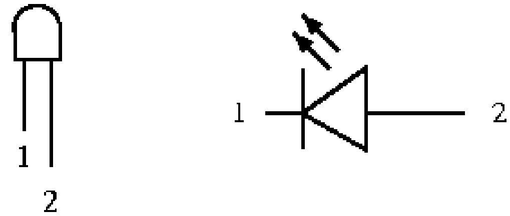


Fig. 4. - LED Pins

Test out an LED:

Apply about 0.7 V to lead 2 (anode) of the LED. Slowly increase the voltage to no more than 1.75. Increase the voltage in 0.01 increments. At what voltage does the LED begin to turn on? Try reversing the polarity of the LED. Does the LED turn on when the voltage is reversed? How is the LED a one-way valve for current?

Using a 1K ohm resistor in series with an LED. Create a V_{LED} vs. I_{LED} graph. Is the LED a linear device? Do not increase V_{LED} beyond 5V in either direction.

2) Creating V_{ref} /2

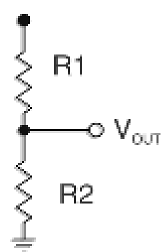


Fig. 5. Resistor divider network

Calculate R_1 and R_2 to have $V_{out} = 0.5V_{cc}$.

Activate power supply voltages and apply 5V to Vcc and use 0V as an input value. If you did your work correctly, you should see Fig. 6. All LEDs should be illuminated.

If your chip is not working, it may be that it is latched onto an erroneous input. If this is the case you must reset the chip by grounding Pin-5 for a second. This will allow the chip to begin sampling the input on pin-6 (Vin).

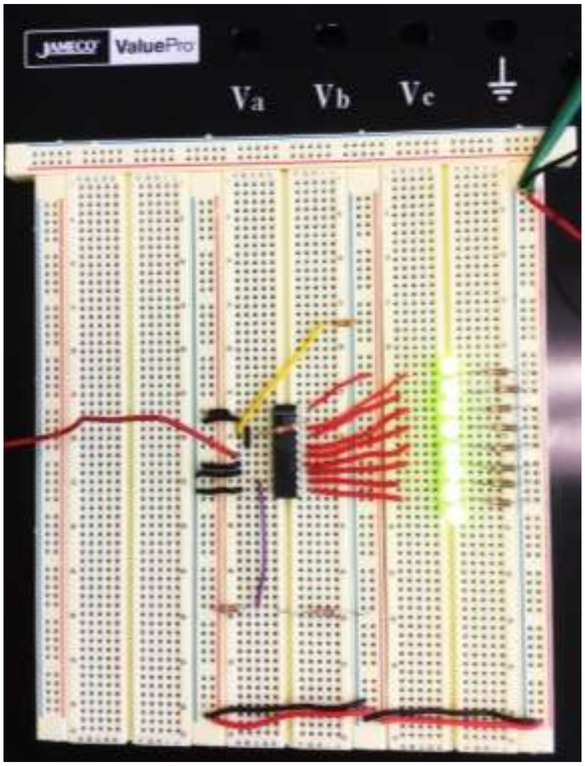


Fig. 6. State of the system with input=0V.

When the input voltage is zero, all the LEDs are 'on' which means the digital output = "00000000". Fig. 6 verifies this state. Now by increasing the input voltage, we can go from 00000000 to 11111111. Using a variety of inputs, record the output codes in 1's and 0's. Write down what the digital number is in base-10. **Fill in the table below:**

Table I. Output Digital Codes

Different Input Voltage Values	Number in base-10	Output Code (binary)
0V		
		00000001
		00000010
		00000011
		01111111
2.5V		
		10000000
		11111110
5V		

Study the figures below. Do you outputs match these digital numbers for the given input voltages? Explain why the values shown below are what you would expect or why you would not expect these values, if that is the case. Show all calculations to explain your answer and reasoning. If your values are close but not exact, explain plausible reasons for this.

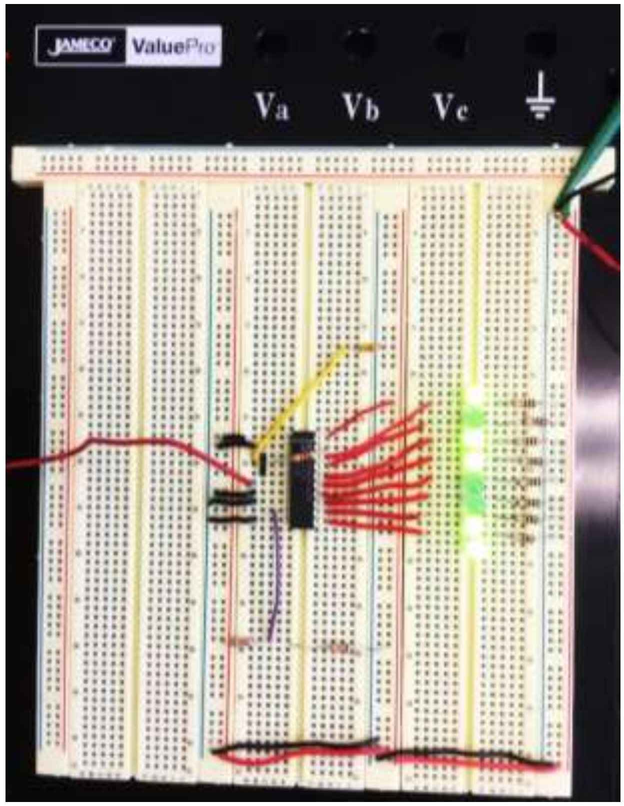


Fig. 7. State of the system with input=1V, output is"00110010".

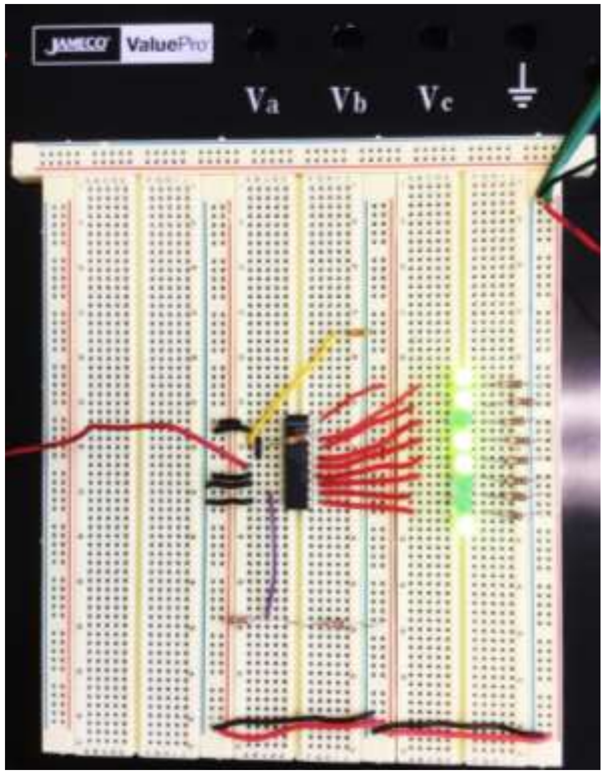


Fig. 8. State of the system with input=2V, output is ”01100100”.

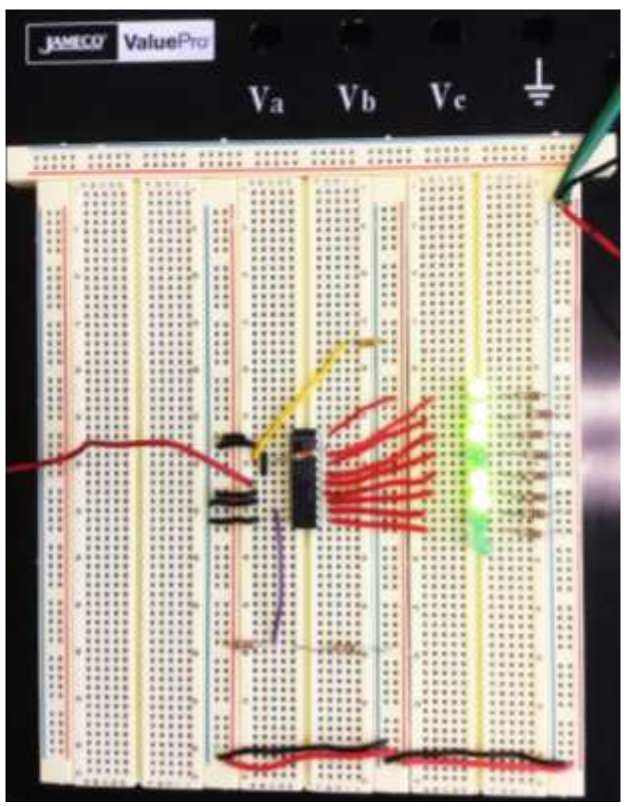


Fig. 9. State of the system with input=4V, output is ”11001000”.

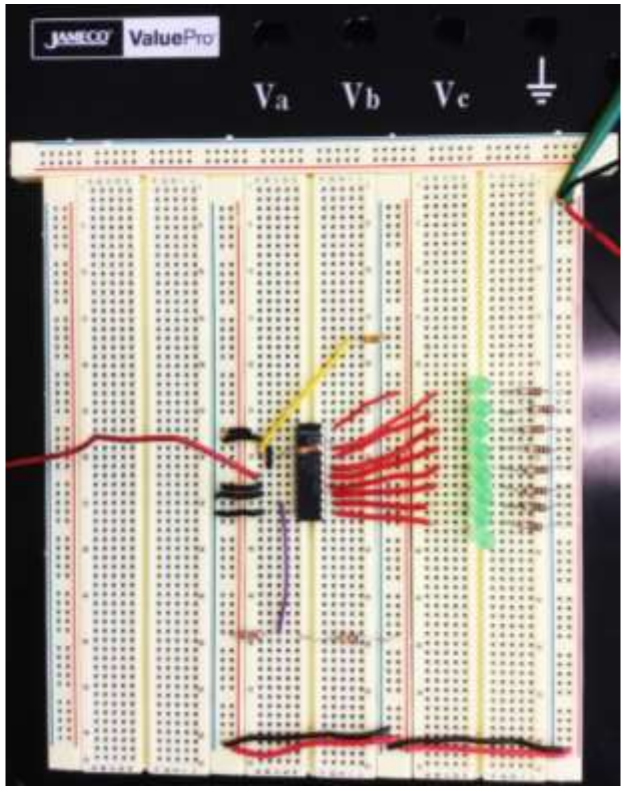


Fig. 10. State of the system with input=5V, output is ”11111111”.

Using a Logic Analyzer

A logic analyzer is an electronic instrument that captures and displays multiple signals from a digital circuit. A logic analyzer may convert the captured data into timing diagrams, protocol decodes, state machine traces, assembly language, or may correlate assembly with source-level software.

Connect the wires to logic analyzer’s connector as shown in the figures below.

Fig. 11. Rigol Logic Analyzer Connector

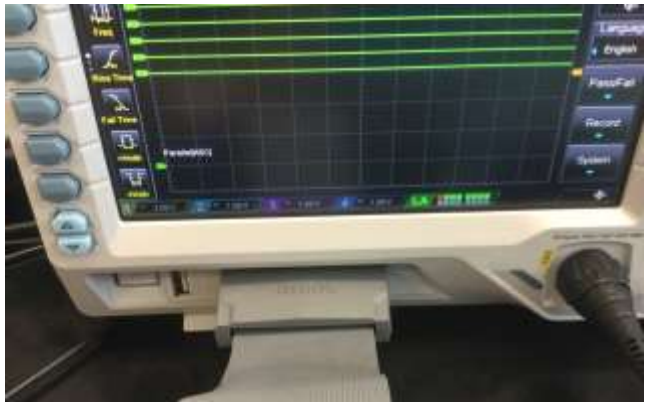


Fig. 12. Connect this connector to the scope’s logic analyzer.

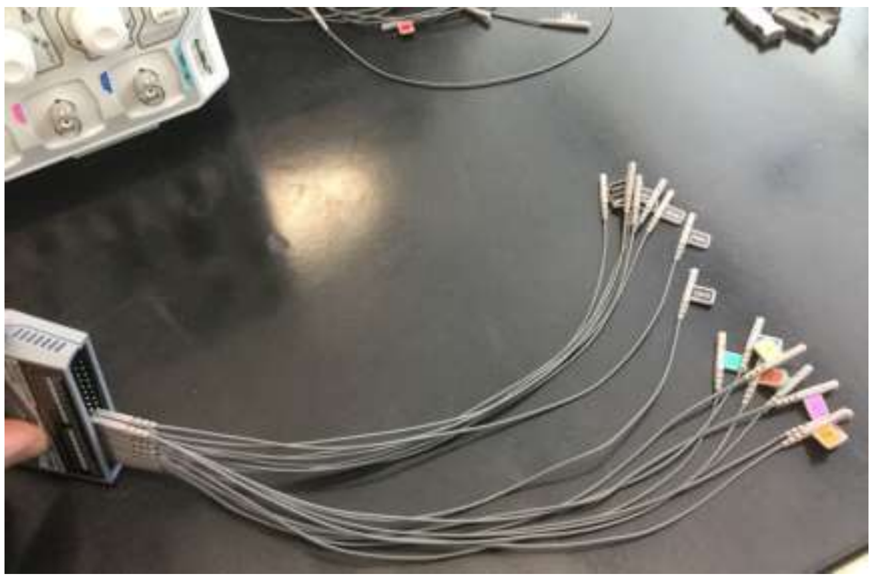


Fig. 13. Plug wires into the Logic Analyzer pins

Fig. 14. Connect each labeled wire to this connector to probe the breadboard.

Next, replace the previous 1K Ω pull-up resistors with new ones (2K Ω) to have correct logic analyzer operation.

Afterward, connect probes LED-by-LED to your system as shown below.

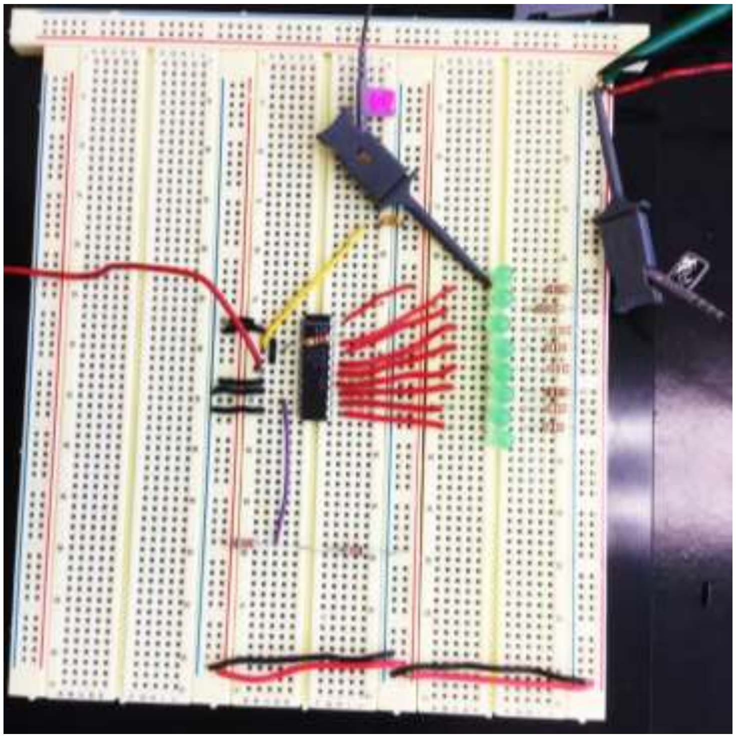


Fig. 15. Connecting one probe to the system.

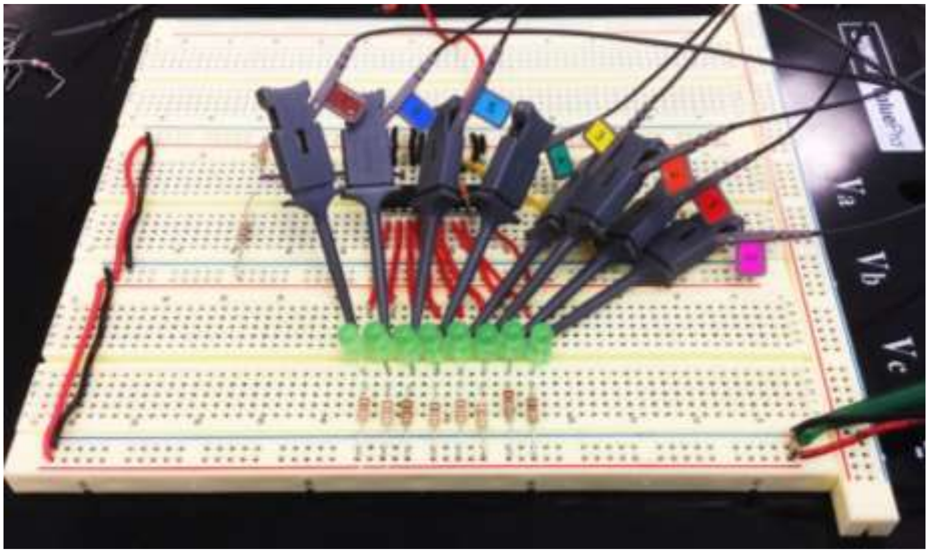


Fig. 16. Whole system after connecting all probes.

Note probe 0 goes to the LSB and probe 7 attaches to the MSB.

Select that you want to view the logic analyzer data on the oscilloscope. Now when changing the input voltage, you should see the digital levels of each pin as shown in the figures below:

Fig. 17. State of the system with input=1V, output is"00110010".

Fig. 18. State of the system with input=2V, output is "01100100".

Fig. 19. State of the system with input=4V, output is ”11001000”.

Fig. 20. State of the system with input=5V, output is ”11111111”.

Using a waveform from the generator to automatically produce a changing input

Create a very low frequency ramp on the generator. Try 100 mHz (0.1 Hz). Use 5 Vp-p with a DC offset of 2.5 V. This will create a continuous input that varies from 0 to 5V and then back down to 0V at a rate of 1V/sec. Use this as your input and observe the LEDs. Do they scroll though all digital values in 5 seconds? Do they count forwards and then backwards? Use the oscilloscope to observe the input.

Next try increasing the frequency to about 200 Hz. You probably will not be able to detect any change in the LEDs that represent the lower significant bits because they will be flickering too rapidly for you to detect.

What to turn in for Part 1:

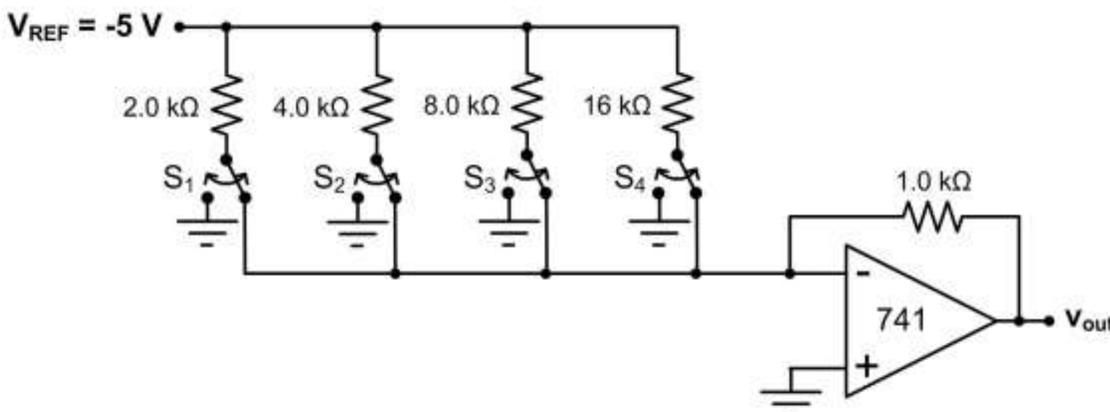
1. Well written report documenting your work.
2. Answer to all questions posed in lab.
3. Data Table
4. Scope screen showing LA data for input values of 1V, 2V, 3V, 4V, and 5V
5. Answer these two questions:
   - Calculate the analog input that would produce an output code of “10001110”. Discuss your answer based on calculations.
   - What is the digital output for 1.15V input? Discuss your answer based on calculations. Does this value produce the digital output you would expect?

ADC LAB - PART 2:

Additional Parts:

1. Two 741 Op Amps
2. Resistors
3. 10K Ω pot
4. Audio cable and separate ear bud cable
5. 5 Resistors that vary by powers of 2 starting with 1KΩ
6. Transducer to use as Vin (Light dependent resistor – Photocell)
7. MC 4511 BCD-to-Seven Segment Driver (one needed)
8. 7-Segment Display (one needed)

Build a 4-bit DAC to convert the top 4 most significant bits of the ADC back into an analog voltage. Create a DAC using an inverting summer, similar to the following circuit:



Your DAC will have 4 inputs and the binary code (S_1, S_2, S_3, and S_4) will be coming from the 4 most significant bits of the ADC chip (pins 11-14). There will be no switches in your circuit because the ADC does this job. Set the resistor values such that each bit is correctly weighted.

Record your output voltage. Use the power supply to vary Vin from 0 to 5 volts.

Compare Vout to Vin. How do they compare? Note that Vout will be inverted compared to Vin. Complete the table below:

Vin	Vout

Using a waveform from the generator to automatically produce a changing input

Create a very low frequency ramp on the generator. Try 100 mHz (0.1 Hz). Use 5 Vp-p with a DC offset of 2.5 V. This will create a continuous input that varies from 0 to 5V and then back down to 0V at a rate of 1V/sec. Use this as your input and observe the LEDs. Do they scroll though all digital values in 5 seconds? Use the oscilloscope to observe the input on Ch1 and the output of the DAC on Ch2. Do you see a range of outputs between 0V and 5V? Sketch what you see. Compare this to the input to the ADC.

Next change the ramp on V_{in} to a sine wave at 200 Hz. Use the oscilloscope to observe both the input and output from the DAC. Try varying the frequency of the input. Try frequencies lower than 200 Hz and higher than 200 Hz. What do you observe? Take a screen shot of the oscilloscope at different frequencies.

How many discrete values do you see on the DAC output? Do you notice more or fewer discrete values (stair steps) as you move to higher frequencies?

How many stair steps should you have in theory?

Can you explain any plausible reasons for why you might see fewer stair-steps at higher frequencies?

How might you increase the speed of the ADC? How might we increase the ADCs sampling rate? What currently is determining the ADCs sampling rate (how often it takes a snapshot of the analog input)? (Refer to the clock info on the ADC’s datasheet.)

Using an audio input

Refer to figure 21 for a complete block diagram of this section.

In this section you will use an audio signal from your cell phone for V_{in}.

First view the audio signal from your cell phone on the oscilloscope. Record the following:

Vp-p (approximately)___________
Voffset (approx.) ______________

Recall that the input to the ADC should be in the range of 0 – 5V. It is likely that what is coming from your cell phone is much less than 5V. Somehow you have to match your audio source to what the ADC expects. You can address this issue is a number of ways: Change the input range for the ADC and/or change the audio signal.

You might want to boost the audio signal and add a DC offset to it. Using 741 op amp ICs, how might you achieve this goal? Design an amplifier that will produce a DC offset such that your audio source spans 0 – 5 volts approximately. Use a potentiometer for the DC input to the amp so that adjustments can be made.

Once your audio is in a form that can be sampled by the ADC, connect the modified audio to V_{in} of the ADC. You should see a lot of flickering LEDs once you do this.

Examine the output from the DAC on the scope and compare it to the ADC's input. How do they compare?

The DAC's output should have a Vp-p of about 5V with a negative DC offset. Use a capacitor to block the DC component.

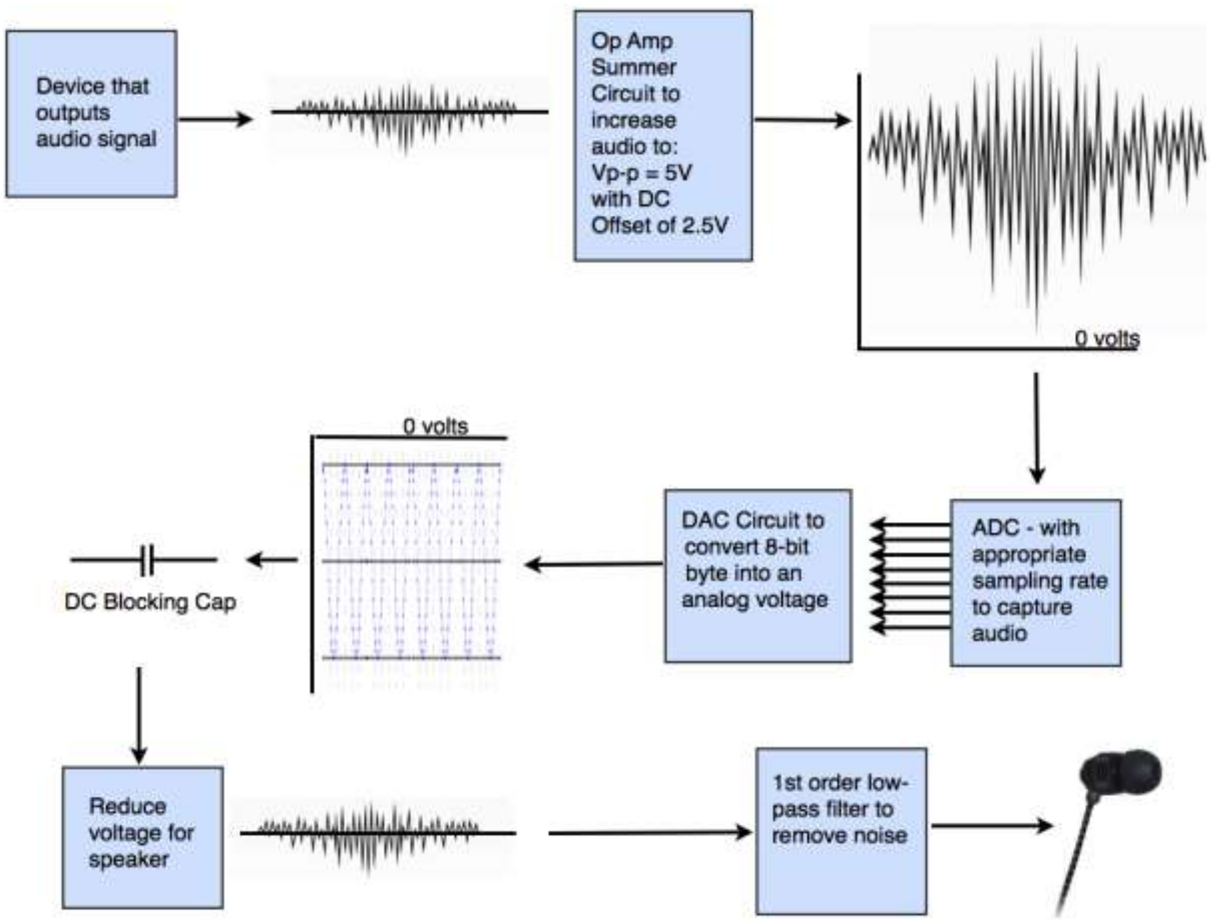
Using an ear bud, listen to the audio signal. This is the audio signal AFTER it has been amplified, sampled, digitized, and then converted back into an analog signal. It will probably sound somewhat distorted but you should easily recognized the audio.

Next using another inverting amp, reduce the audio to a lower voltage level so that it is closer to the voltage level of the original signal.

How to make the audio sound better? Try increasing the sampling rate of the ADC by lowering the capacitor in the RC network between pins 4 and 19. Using a first-order low-pass filter, try removing some of the high frequency noise.

Also you might want to connect more bits to the DAC so that there are more stair steps in the DAC output. **Before changing anything, probe your circuit with a scope probe to determine where distortion seems most pronounced. Try to address the big issues first.**

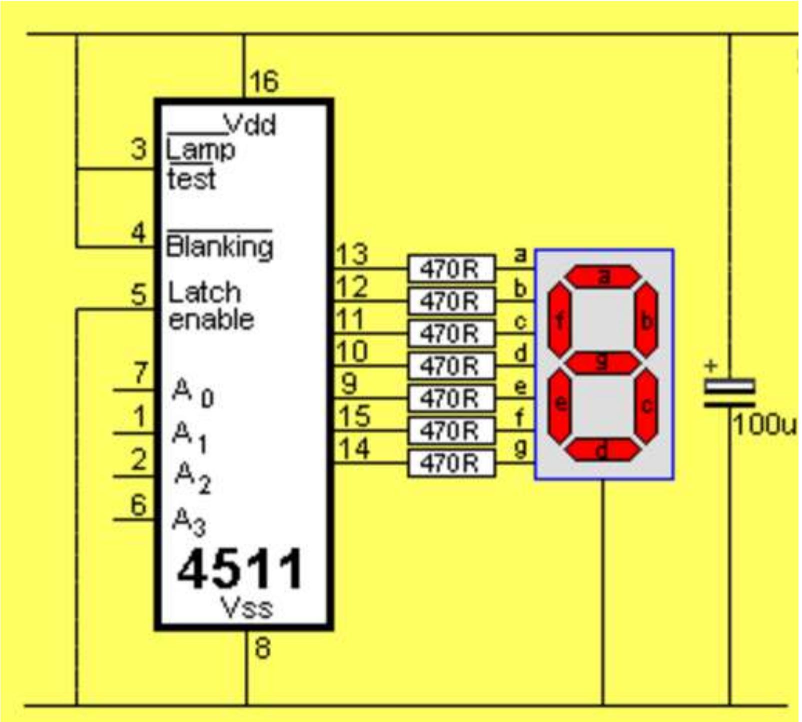
Figure 21 - Block Diagram for Audio to Digital to Audio

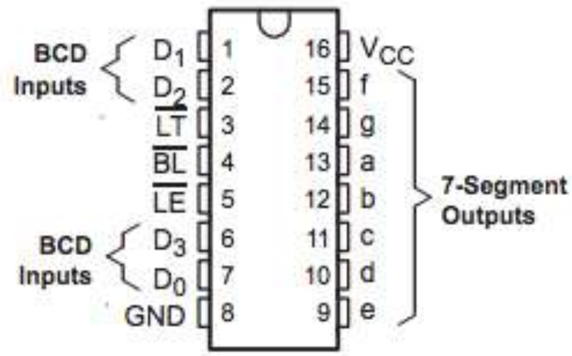


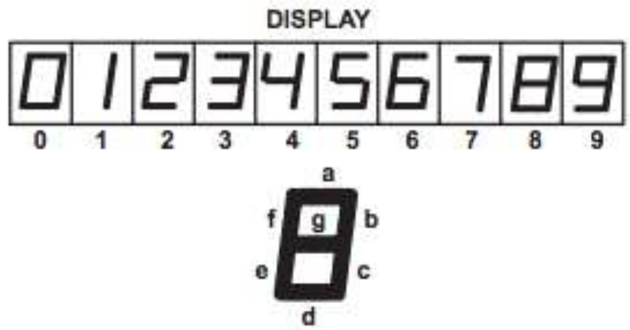
Creating a base-10 number to represent the output of the ADC

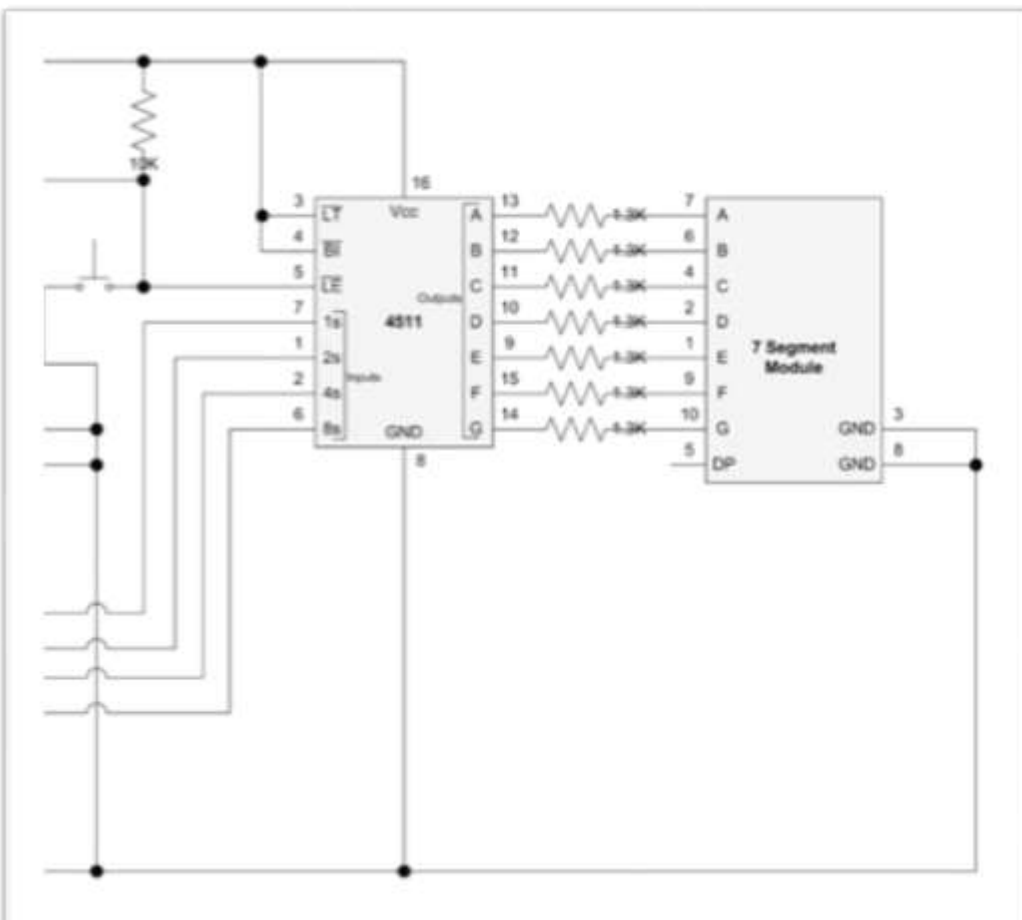
Next you will create a base-10 single digit output that represents the digital code from the ADC. You will use a MC 4511 BCD-to-Seven Segment Driver and one 7-segment display.

Your circuit should look something like: (Note Vdd = 5V and Vss=0V)

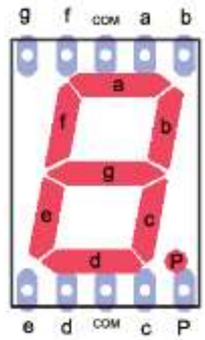



BCD Inputs
D1
D2
LT
BL
LE
BCD Inputs
D3
D0
GND
VCC
f
g
a
b
c
d
e
7-Segment Outputs


DISPLAY
0 1 2 3 4 5 6 7 8 9
a
f
g
b
e
c
d


10K
Vcc
4511
Outputs
Inputs
GND
A
B
C
D
E
F
G
DP
7 Segment Module
GND
GND

The pin out for the 7-segment display is:



How accurate is the seven-segment display at representing the 4-bit number? What codes can not be represented with the 7-segment display?

Next using a Cadmium-Sulfide photocell, create Vin such that the input depends on the light falling on the light sensitive resistor. You will want to measure how the resistance changes in the photocell to determine how you will use it. Find the range of resistance and record the range.

Set V- to the lowest Vin value from the photocell. Set Vref = ½ * range of Vin. Use a pot for Vref so it's value can be adjusted later.

Draw the circuit you are using to create Vin with the photocell:

How does the ADC perform with the photocell?

Is there a way to "calibrate" the ADC such that the ADC only produces 0000 to 1001 for the full range of the input?

What to turn in for Part 2:

Answer all questions by re-writing the questions and typing answers in a write-up. Include tables where needed.

Sketch all circuits built and show waveforms/outputs.

Find a consumer electronic device with ADC and DAC circuits within it. Explain at least two benefits of using these circuits. When was the concept of digital information invented?