Purdue University
School of Mechanical Engineering

ME 586: Microprocessors in Electromechanical Systems

Fall Semester 2022
Dr. P. Meckl

Laboratory Assignment #6

Objectives:

1) To Construct Digital-to-Analog and Analog-to-Digital Converters
2) To Perform ADC and DAC Operations on the STM32 Microcontroller

Procedure:

During the one-week period of this lab you will construct analog-to-digital and digital-to-analog converters.

Task 1: Construction of Digital-to-Analog Converter (DAC)

Using the DAC0808 8-bit DAC chip and LF353 op amp (both integrated on the supplied DAC board), you will need to generate analog output voltages V_{out} for an equally-spaced subset of digital inputs between 0 and 255.

In order to connect the DAC board to your hardware, use the connections shown in Figure 2. **DO NOT TURN THE POWER SUPPLY ON UNTIL YOU ARE SURE YOU HAVE CONNECTED THE BOARD CORRECTLY. IT IS VERY IMPORTANT TO CORRECTLY CONNECT THE POWER SUPPLY LINES, OTHERWISE YOU MAY DAMAGE THE DEVICE.** MSB corresponds to the most significant bit and LSB corresponds to the least significant bit. Vo is the output voltage, Gnd is ground, and the +15V, -15V, and +5V supplies should be taken from the breadboard supplies on the microcontroller box.

First connect the digital inputs with switches on the microcontroller box and test the behavior. Once the circuit works, use the STM32 and appropriate software from Homework #5 Problem 4 to generate the 8-bit digital values. Use the multimeter to record the resulting voltage. Plot the results (voltage vs. digital value) and comment on offset and linearity errors (if any).

Checkpoint 1: Show your plot to the TA

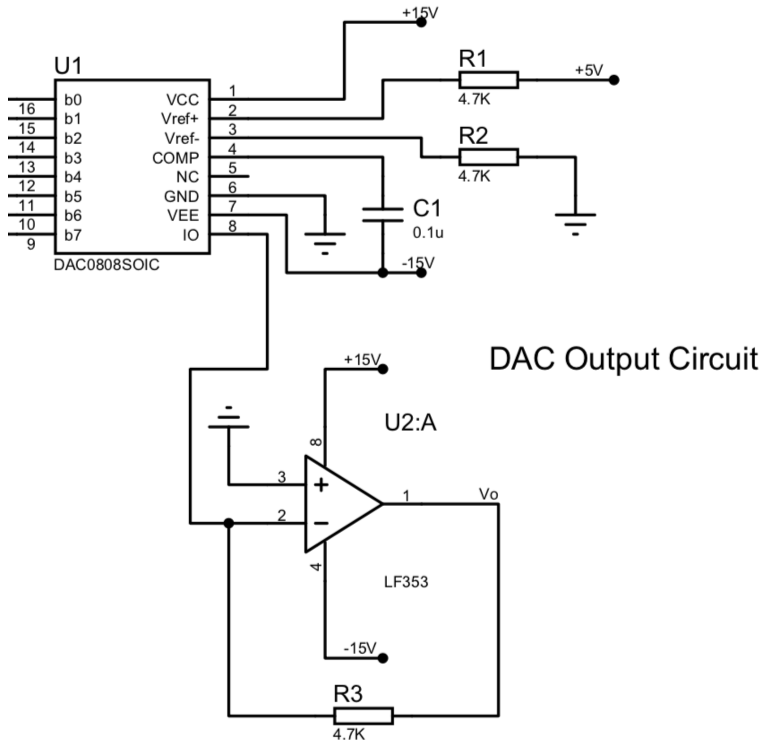

Figure 1: 8 Bit D/A Converter Wiring Diagram

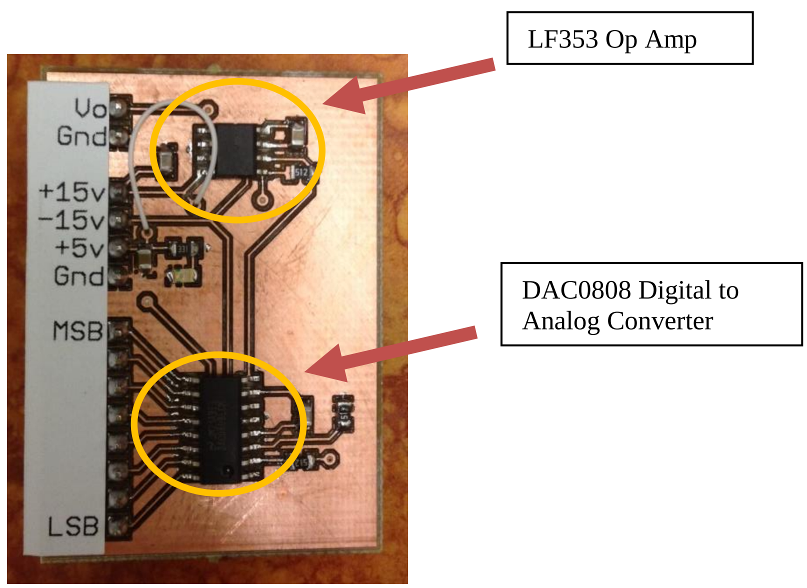

Figure 2: DAC Board

Task 2: Construction of a Comparator

This task is to construct a comparator with an LM311P comparator chip and a 1kΩ pull-up resistor. Make the connections shown in Figure 3. Use the HP/Agilent Power Supply to generate a constant analog voltage. Set the output of the power supply to 2.5 Volts and connect it to the analog input (pin 3) of the comparator. Connect pin 2 of the comparator to ground and check the output voltage of the comparator (pin 7). Now connect pin 2 to the 5V supply line and check the output voltage of the comparator (pin 7). What voltages did you get for logic HI and logic LO at pin 7? Now remove the 1kΩ pull-up resistor and measure the voltages for logic HI and logic LO at pin 7. Is this voltage range appropriate to use with TTL logic chips? Comment on the use of pull-up resistor to shift the output range. Now the re-connect the pull-up resistor so that you can perform the next task.

Checkpoint 2: Show your working comparator and explain why the voltage range is shifted.

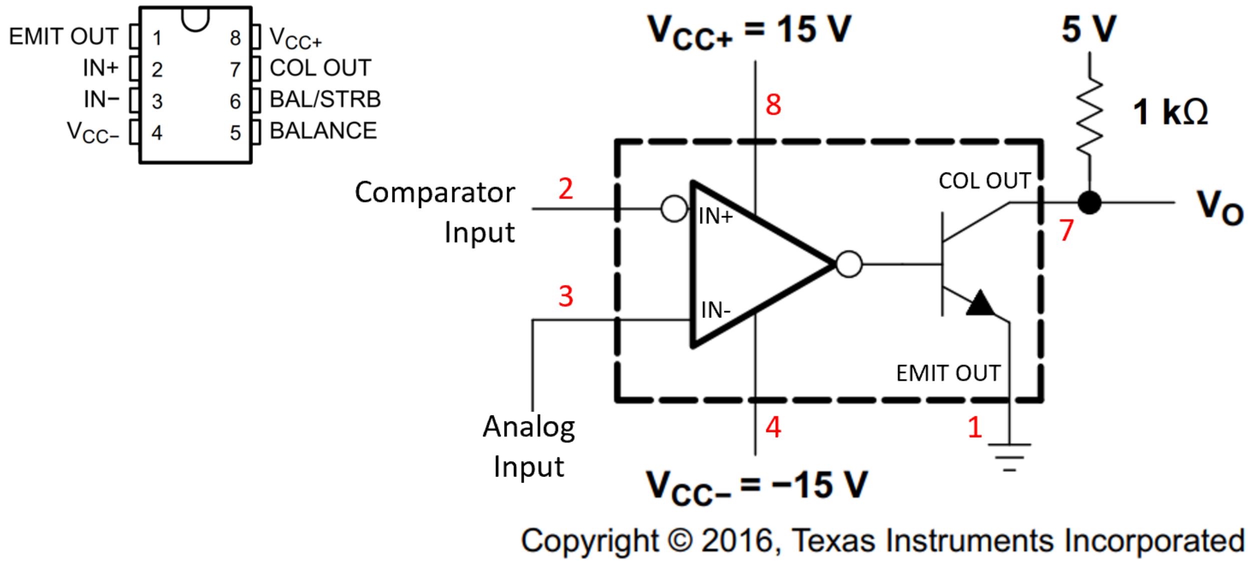

Figure 3: LM311P Comparator Chip wiring diagram

Task 3: Construction of Analog-to-Digital Converter (ADC)

As covered in the lecture, most ADC's consist of a DAC, a comparator, and a counter. Use the DAC0808 of task 1 and comparator of task 2 to construct the system of Figure 4.

The microcontroller serves as a counter. During each cycle, it sends a digital output value of the counter. This digital value is sent to the input of the DAC, using the digital output port pins. The resulting analog output is sent to a comparator and is compared with the analog input signal. Comparator output is sent to the digital input port of the microcontroller. When the DAC voltage exceeds the analog input voltage, the comparator output goes hi to inform the microcontroller to stop counting and to display the digital value. You should use an external key press to start the conversion (SOC).

Use the software you developed for the STM32 in Homework #6. Just change the logic so that the counter increments as long as the microcontroller receives a zero at PC6 and the counter resets and displays the digital value when it receives a 1.

After construction of your ADC system, perform the following:

i. Input constant voltages from 0 to +5 volts using the HP/Agilent power supply. Find the corresponding digital values and plot your results. Use Oscilloscope to see output voltage. Vmax should correspond to your input voltage, with small quantization errors.

ii. Use the function generator to input a sine wave to your ADC system. Perform the conversion and display the outputs of the DAC and the input sine wave on the oscilloscope. Use both low and high frequencies for the sine wave. Comment on the results.

Checkpoint 3: Display the output of DAC and the input sine wave in the oscilloscope and explain the differences between using low and high frequencies for the sine wave.

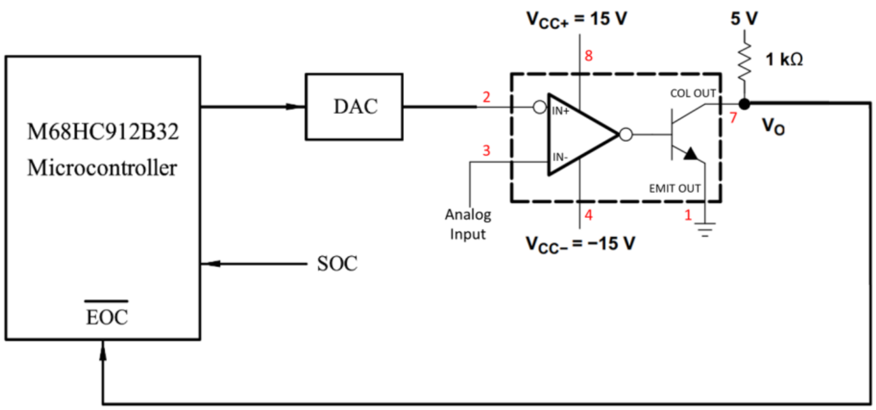

Figure 4: ADC wiring diagram

Task 4: Analog I/O on the STM32

The STM32F100RB microcontrollers in the lab are equipped with 6 channels of analog input (using an analog multiplexer and an ADC with 12-bit resolution) and two channels of analog output (also 12-bit).

Analog I/O on the STM32:
The STM32 box provides 6 channels of analog input and 2 channels of analog output. BNC connections are provided on the STM32 box for access to these signals.
The voltage range for the analog inputs can be individually set with a switch to ±5 V (10 Vpp) or ±10 V (20 Vpp). The analog outputs provide signals within ±10 V so set the switch accordingly.

The C functions `initadc`, `initdac`, `a_to_d` and `d_to_a` are available in the library.

<u>DAC Calibration</u>:

To begin, develop a simple C program that sends 20 values equally spaced from 0 to 4095 to D/A channel 0 (AO0), waiting after each one for you to press a key. This gives you time to record the output voltage on channel 0. Plot the voltage vs. DAC counts and comment on offset and linearity.

ADC/DAC Experiment:

Write a simple C program that reads in a voltage on A/D channel 0 (AI0) and sends the corresponding digital value to D/A channel 0 (AO0). Construct the experiment illustrated in Fig. 5, using a function generator and an oscilloscope.

Compare the resulting output sine wave with the original input waveform on the oscilloscope. Increase the frequency of your sine wave and comment on how the output waveform changes with frequency.

Checkpoint 4: Display your result and explain the difference between using low and high frequencies for the sine wave.

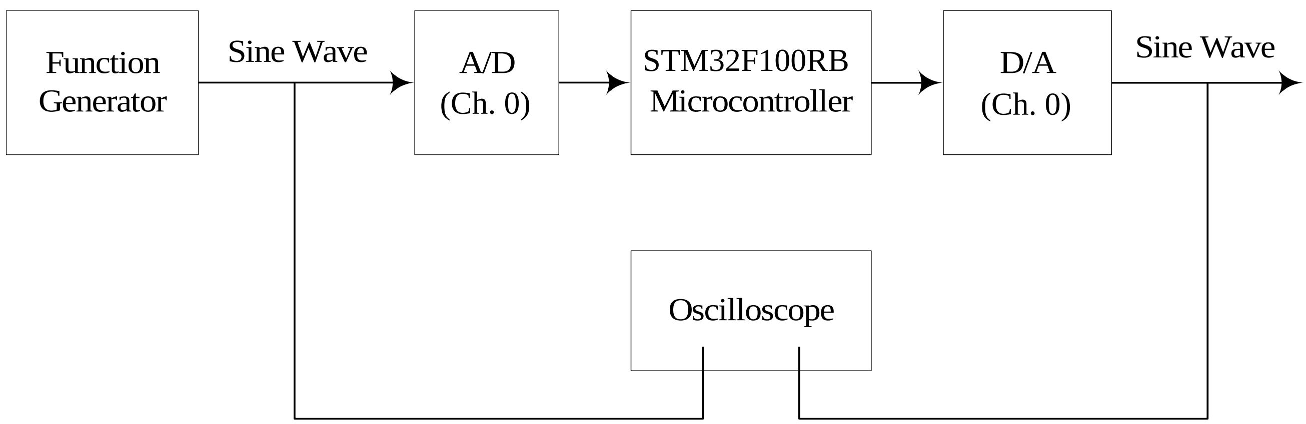

Figure 5