AD - DA Conversion Manual

Jona Bühler

ETH Zürich

February 18, 2019

Abstract

The goal of this experiment is to develop a complex mixed-signal circuit from scratch. The circuit will be able to convert analog signals to 3bit digital signals, to represent them as a binary number series (AD conversion) and to convert the digital signals back into analog ones (DA conversion). Using the digital storage oscilloscope the analog input signal is visually approximated by a step function. During the process you will get to know operational amplifier, comparator and priority encoder ICs. Finally, the converter circuit is tested in various frequency regimes to investigate its possibilities and limitations. Those limitations are then connected to the properties of the components in use and general theoretical limits such as the Nyquist-Shannon-Sampling Theorem.

Contents

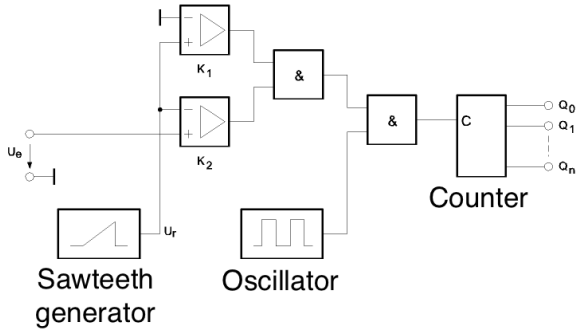

Figure 1: Working principle of a ramp converter. Modified from [6]

1 Introduction

When performing a measurement, the result is usually an analog signal, as for example a change in voltage of some circuit. In order to further process the signal it is often convenient to convert the analog signal into a digital signal. This conversion can result in loss of information as an analog signal (e.g. voltage) can assume any value, but the digital signal is discretized. Additionally, a conversion itself puts a limit on the sampling speed of a signal. Depending on the application, various conversion methods exist. Three basic AD conversion methods are briefly described here, and the last method is the topic of this experiment.

Ramp conversion The first method that we will discuss is the so-called ramp converter or sawteeth converter. In Figure 1 a schematic of its main components is shown.

The main idea here is to compare the input signal U_e to the constantly increasing ramp voltage U_r and the ground voltage by the comparators K_1 and K_2. By doing this, the first AND gate will output logic one as long as $U_r < U_e$ and after that logic zero. The second AND gate is connected to a quarz oscillator and the output of the first AND gate, which means that it will switch between one and zero for each period of the oscillator but only until

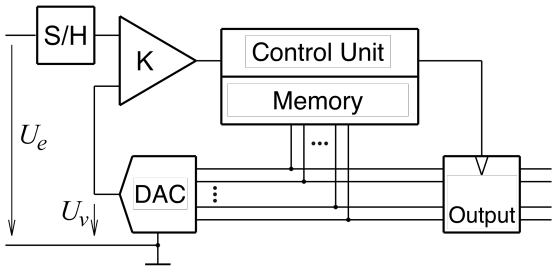

Figure 2: Working principle of a feedback converter. Original [7]

$U_r > U_e$. These switchings are then counted and since we have a constant frequency of the oscillator and a constant increase of the saw teeth voltage the number of counts is in the end proportional to U_e. This linear relation can be calculated in advance which allows us to associate a certain number of counts with a corresponding binary number. One of its advantages is clearly that we can increase the conversion resolution without adding any more parts to the circuit but just by increasing the oscillator's frequency and adjusting the 'count number' to 'binary number' relation. On the other hand the input signal must not change quickly since the highest time resolution depends on the period length of the sawteeth signal. This converter is part of the category called integrating converters or counting methods since a counter is used in the last step. [5]

Feedback Converter The second method presented here is the feedback converter, schematically shown in Figure 2

For this method we do not directly convert the input signal but assume succesively increasing (decreasing) binary values in a certain way. The assumed digital value is then converted with a Digital-Analog Converter (DAU in the Figure; how this works is explained in section 2.4) and compared to the original input signal (by the comparator K in the Figure). As long as the input signal is higher (lower) than the assumed value the process is repeated with the next higher (lower) assumed digital value. But since the process of assuming and comparing takes some time until the right value is found we need a so called Sample-and-Hold-circuit (S/H in the Figure) that leaves the input signal on a constant level during the conversion. There are different techniques of how to approximate the signal successively and certain methods reach a 16-bit resolution at a converion rate of $1MHz$. But their implementation is way more difficult and not presented here. For further information you are encouraged to have a look at [5].

Flash conversion The last method, which is the circuit we will implement here, is called a **flash converter**. Its general components are:

- a linear voltage splitter,
- comparators,
- a priority encoder,
- a LED display.

The main idea is to use equally seperated reference voltages and compare them to the input signal. This discretizes the signal with a certain n-bit resolution. The discretized signal can then be converted into binary code. In section 2 the whole concept and the single components will be explained in more detail. It is the easiest way to implement AD conversion and part of the category called 'parallel converter' since several comparators are used at the same time. Only the propagation delay times of the gates themselves determine the conversion speed. This makes it really fast (nanosecond discretization; this also explains the name 'flash converter') but for each bit more in amplitude resolution the practical expense increases exponentially. This limits its practicality in terms of size and resources. The basic concept is based on [1].

Sampling theorem The speed of a conversion process is obviously crucial for the quality of the converted signal. Indeed there exists a certain sampling frequency for each signal such that from the discretized digital signal one can reconstruct the original analog signal without any loss of information. This basic and important statement in its complete version is known as the **Nyquist-Shannon-Sampling-Theroem** and states:

Theorem.
Nyquist-Shannon-Sampling-Theroem

If a function $x(t)$ does not contain any frequencies higher than B Hz*, it is completely determined by giving its ordinates at a series of points spaced* $\frac{1}{2B}$ *seconds apart. [8]*

From this follows that when sampling an analog signal with maximal bandlimit B Hz a sufficient sampling rate would be anything larger than 2B Hz The factor two is called the 'sampling rate factor'. On the other hand, a sampling larger than 2B Hz does not provide any more information about the original signal. Thus, the bandlimit already defines the requirements on the speed of the conversion. As an example let us consider an Audio CD: The highest frequency that the human ear can hear is around 20 kHz. According to the theorem we therefore have to sample the signal with a rate of 40 kHz. But we also have to be prepared for the case when the signal contains higher frequencies because if not, when reconstructing the signal, this results in so called Aliasing-errors. These errors occur due to the ambiguity of the sampled signal and result in frequencies that were not part of the original signal. To avoid these errors one filters the audio signal with a low-pass filter before converting it. But due to the finite slope of such a filter it does not cut off all frequencies higher than a certain limit but the ones close to the limit are only diminished. Because of this the sample rate factor is set to ≈ 2.2 instead of 2, resulting in a sampling rate of 44.1 kHz. [8]

2 Methods and Experimental Setup of the AD-DA converter

In this section we will develop the AD-DA converter step by step to obtain a functional circuit. You will be asked to implement this circuit in section 3.

2.1 Devices

For the experiment we use the following devices:

- A breadboard, cables, resistances and ICs to realize the circuits;

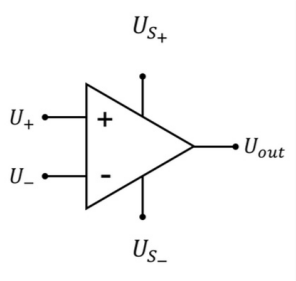

Figure 3: The Symbol of an OpAmp or a Comparator [10]. U_+ and U_- are the inputs whereas U_{S_+} and U_{S_-} are the supply voltage connections. U_{out} is the output.

- a function generator with variable frequency to generate the signal;
- a digital storage oscilloscope (DSO) to display and save signals and waveforms;
- a variable DC Power Supply with 3 Channels to supply the different circuit components with the desired voltage;
- Two multimeters to check the functionality of the circuit at various points.

2.2 Basics

In this section some basic circuits and concepts will be introduced that will be of importance later on.

OpAmp The concept of an operational amplifier (OpAmp) should be known from Physics Lab 1 and 2. Make sure you are familiar with the terms positive and negative feedback, the 'golden rules of OpAmps' and the idea of inverting and noninverting amplification [10]. The Symbol that represents an OpAmp is shown in Figure 3.

Comparators A comparator is the smallest unit to convert an analog signal into a digital one. It is a circuit with two inputs and one output. The input that is labelled with '+' is the non-inverting input and the one labelled with '−' is the inverting input. It fulfills the purpose to output the positive supply voltage in case the voltage connected to the non-inverting input is higher than the voltage connected to the inverting input, and it outputs the negative supply voltage otherwise. It is basically an

OpAmp but without frequency compensation in order to achieve a higher edge steepness. Because of this it is not possible to use feedback. OpAmps and Comparators share the same symbol (Figure 3).

Table 1: Input-Output relation of a comparator

Input	Output
$V_+ > V_-$	1, high
$V_+ < V_-$	0, low

For the purpose of our circuit it is important to note that there also exist other types of outputs, so-called 'open-collector outputs', which output different voltages than explained above. This is treated in the next section. However, the logic principle behind it is still the same (see Table 1). [9]

Open-Collector Output The usual way to realize an output is the push-pull output (PPO). Figure 4 shows the schematic of a push-pull output stage on the left side.

Depending on the IC output, either one of the two transistors T_1 or T_2 receives a voltage at its base. In the case of T_1 the transistor connects the positive supply voltage of the IC to the output. In the case of T_2 the same happens with the negative supply voltage. A problem arises when we try to connect two push-pull outputs PPO_1 and PPO_2 together. Assume that PPO_1 has output V_+ and PPO_2 has output V_-. By connecting both we create a short-circuit, which is in general bad. The problem can be solved by adding additional components such as diodes. Another possible problem is that the high and low voltage output levels are already defined from the beginning by the supply voltage of the IC. In fact, in our AD-DA converter circuit we will encounter the problem that we need different voltage levels at the output than at the input.

A way to solve the latter problem is by introducing the open-collector output, presented on the righthand side of Figure 4. The difference is that T_1 is missing and we added an external, so-called pull-up resistance R_1 and an external voltage source U_{out}. If the IC output is 1, the transistor T_3 will not connect through so at the output we get U_{out}. The resistance R_1 limits the current. If the IC has output 0 the transistor T_3 will connect the IC's GND through such that the output will be on the GND level. Note that with using this type of output stage we solved both problems mentioned above, i.e. it is now possible to connect several outputs together without creating a short-circuit and we have an independent positive output voltage level.

Pre-resistances of LEDs A LED is a diode that emits light when supplying the right voltage to it. The operating voltage of an LED depends directly on its colour (via the wavelength), whereas the current controls the intensity of the light. The operating voltage is thus depended on which LED we use and is not manipulable, but the current is manipulable. To limit the current and to control the intensity we have to introduce a so-called pre-resistance. Moreover, when we use the LED in a circuit it is often the case that the operating voltage of the circuit is higher than the operating voltage of the LED. The pre-resistance also takes care of that issue.

Exercise 1

The goal of this exercise is to calculate the required resistance for the LEDs that will be used in the converter later.

- Get yourself three (equal) LEDs from the Help Desk. Three, because you will need three later on.
- You can find the optimal working voltage U_{LED} in the datasheet of the LED.
- Our circuit will work with $U = 5\,\text{V}$. Calculate $U_R = U - U_{\text{LED}}$.
- A LED works perfectly fine for a current of $I_{\text{LED}} = 10\,\text{mA}$. What resistance do you need? Choose the next highest resistance that is available when implementing the real circuit. (Note, that a common mistake is to operate electronic circuits at their maximally allowed input current. This, however, only shortens the lifetime of the component and is (almost) never necessary.)

Exercise 2: Analog to digital converter

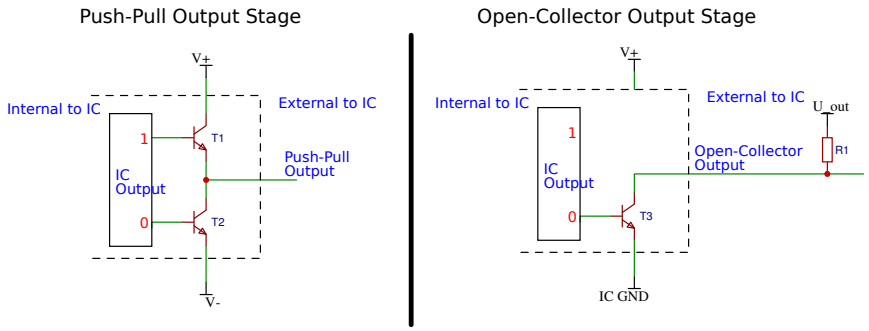

Figure 4: A push-pull output stage (left) vs. a open-collector output stage (right). Replacing T_1 by R_1 allows us to implement a positive output voltage which is different from the IC's supply voltage. In this way we may also connect several outputs together without short-circuiting the system.

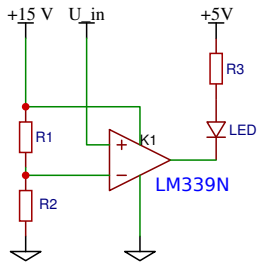

Figure 5: A 1bit analog to digital flash converter.

The goal of this exercise is to conceptionally understand the smallest AD flash converter that is possible. Implement the circuit shown in Figure 5. Choose R_1 and R_2 such that the voltage at the inverting input of the comparator LM339N is 7.5 V and the current does not exceed 10 mA. The resistance R_3 is the one you calculated in exercise 1. Use the function generator as input source. Make sure that the input signal is between 0 V and 15 V. Try out what happens for different functions and frequencies. Also connect the DSO in a way that allows you to compare the input and output voltages.

2.3 The 3bit AD Flash Converter

2.3.1 The idea

As already mentioned in the Introduction the circuit consists of four parts: the linear voltage splitter, the comparator section, a priority encoder and LEDs.

In Figure 6 the overall idea is presented schematically: In Exercise 2 we compared the signal to half of the reference voltage while now we want several comparators to compare it to different voltages to get a higher resolution. These equally spaced voltages will be provided by the linear voltage splitter. All comparator outputs will then be combined and converted to a binary number inside the priority encoder. Its 3bit output is then connected to three LEDs via pre-resistances or to the DA converter as explained later.

The circuit realized in this work is able to convert only positive voltages and the externally provided reference voltage V_{ref} defines the upper limit, thus only signals in the interval $[GND, V_{ref}]$ can be converted. It is clear that a discrete spectrum of values can only approximate a continuous one so the second thing to think about should be the resolution of the conversion. For example, to approximate a function with discrete values the concept of a step function is very useful. We chose to split the range into 8 discrete value (including 0) such that the conversion could not only be displayed as a step function but also as a 3bit-binary number (counting from 0 to 7) by LEDs.

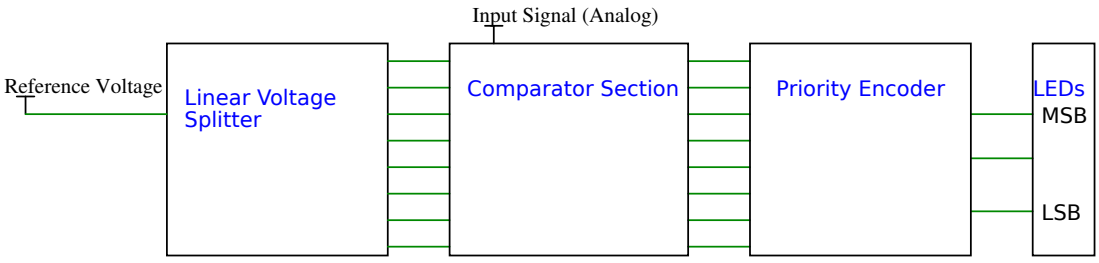

Figure 6: This scheme visualizes the basic idea of the AD flash converter.

2.3.2 Implementation

The second step to think about is how to experimentally implement this idea. We split the above problem into three smaller problems:

1. Discretization
2. Conversion into the binary number
3. LED display

Discretization: The discretization process is taken over by a combination of a linear voltage splitter (indicated as R_1 to R_8 in Figure 7) and a band of comparators (K_1 to K_7). To work properly the comparator (see section 2.2) had to be connected to 15 V supply voltage and 0 V GND. Thus, the output would exceed the logic level of 5 V by 10 V. As explained in section 2.2 we overcome this problem with so called 'open collector outputs'. The external voltage level can then be set to 5 V. To limit the current we added the pull-up resistances R_{12} to R_{18}. Now that we made sure that the ouput is on a well defined logic level we may combine the linear voltage splitter and the comparators. This happens by taking a so-called reference voltage V_{ref}, defined as the maximum voltage of the input signal and connect it to the linear voltage splitter. After the k-th resistance the voltage will have decreased by

$$V_k^{\text{decreased}} = (\sum_{i=1}^{k} R_i) \cdot I = \frac{(\sum_{i=1}^{k} R_i) \cdot V_{\text{ref}}}{(\sum_{i=1}^{8} R_i)} \quad (1)$$

since I is constant in the whole reference circuit. This defines the height of our discretization steps. We choose R_1 and R_8 to have half the value of the resistances R_2 to R_7 such that the lowest and highest levels are well defined (i.e. there is no ambiguity in how those input levels are represented).

Exercise 3

At this point, think about how the discretization step size depends on the values of the surrounding resistances. Make a table that presents the relation between the output signal and what information we obtain from it about the input signal with certainty (e.g. if the output signal is 1-0-1, we know with certainty that the input signal is between .... and .... Volts.) If you want, you can also think about how the circuit can be modified such that all steps have size 1 V and are still well defined.

So the actual voltage after the k-th resistance will thus be

$$V_k = V_{\text{ref}} - V_k^{\text{decreased}} = \sum_{i=k+1}^{8} R_i \cdot V_{\text{ref}} \quad (2)$$

for $k = 1, 2, ..., 7$. Now we can simply compare the input signal to V_k using 7 comparators. To implement this we connected the input signal to the non-inverting input of the comparators. At its inverting input the comparator K_{8-k} receives its own reference voltage V_k. In this way the (8-k)-th comparator output will remain at logic zero as long as the input signal does not exceed V_k. Notice here that we do not need a comparator to detect the 0 V level but only as soon as the signal rises above that, which is why we need 7 and not 8 comparators for a 3bit conversion. In this way we are able to discretize the input signal.

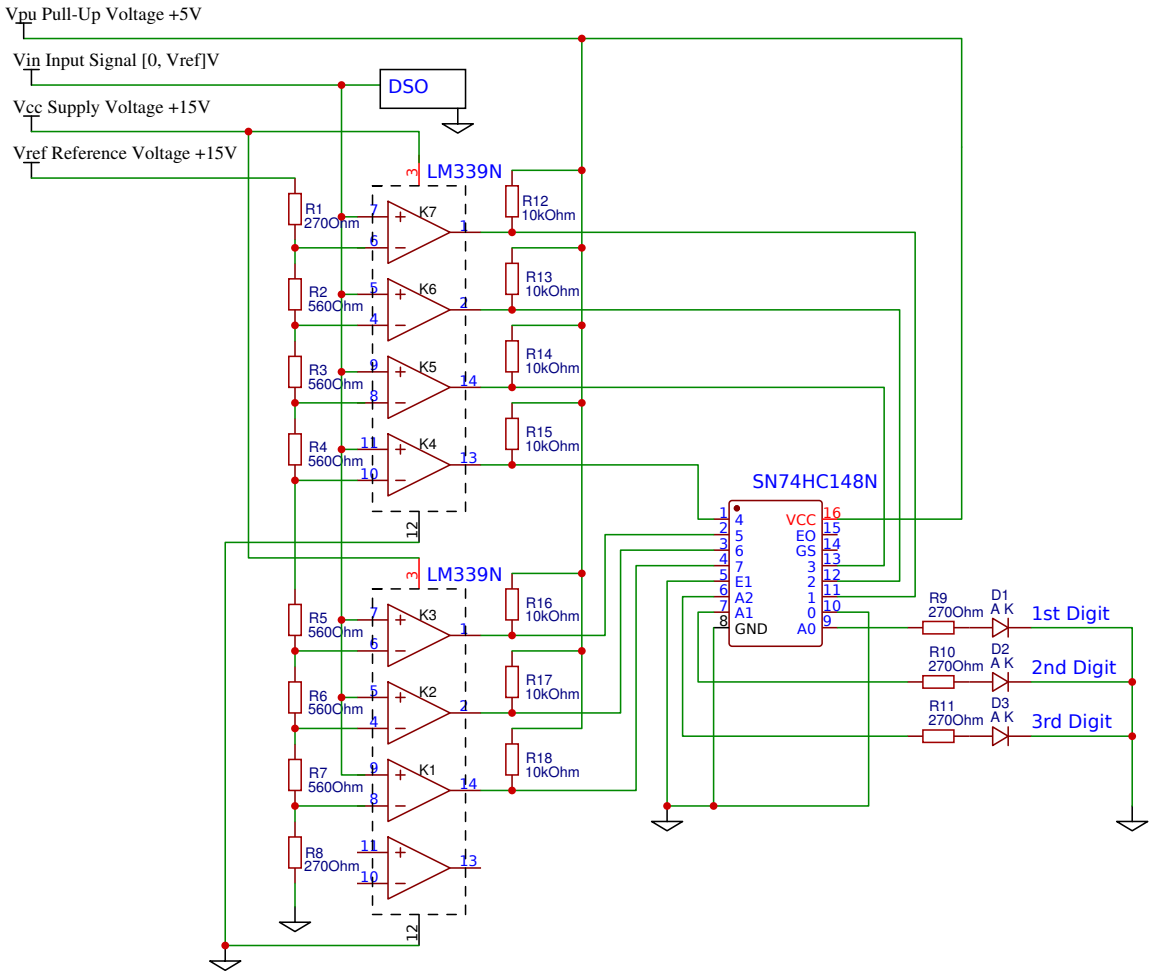

Figure 7: The 3bit Analog-Digital Flash Converter Circuit. From left to right there are all used voltage sources, the linear voltage splitter, the comporator section, the pull-up resistances, the priority encoder and the LEDs. For further explanations see section 2.3.

Conversion into binary number: What we have now is a series of logic outputs that are priority decoded. What does this mean? Imagine comparator K_4 to output a logic zero and comparator K_3 to output a logic one. This tells us immediately that K_2 and K_1 also must output a logic one and the comparators K_5, K_6 and K_7 a logic zero, just beacuse of the way we connected them to the voltage splitter. This way of reasoning about the state of the other comparators just by knowing the highest index l for which the comparator K_l has output one is what we call here 'priority decoded outputs' or 'priority code'. It is a so-called summed 'one-hot code'. So what we need is a converter from priority code to binary code, i.e. a logic circuit that inputs 8 logic values and depending on how many of them are set to logic one it outputs its 3bit binary equivalent. This may be obtained by a series of logic OR gates that form a complex logic circuit. This kind of circuit that fulfills exactly the requirements mentioned above is available as an IC gate, called 'Priority encoder'. For further information about this logic IC we refer to its datasheet [3] and Tietze & Schenk [1].

LED display: The three outputs of the priority decoder represent the 3 digits of a 3bit binary number. Each digit can only take the values zero or one and so can the outputs. This suits the idea of representing the digit zero with an LED turned off and the digit one with an LED turned on. For this purpose we simply connected the outputs via pre-resistances (that decrease the 5 V output voltage down to the working voltage of the LED) to LEDs and then to the ground potential.

2.4 The 3bit DA converter

Exercise 4: Digital to analog converter

To enter the topic of DA conversion we start by implementing and understanding a 2bit DA converter. Take a look at Figure 8. Your task is to implement the circuit that is presented in the Figure but without the whole branch which includes R_1 and R_2 and is connected to input 6 of the priority encoder. In our case the branch that leads to number 7 represents the most significant bit (MSB) of a 2bit binary number and the branch that leads to number 9 represents the least significant bit (LSB) of a 2bit binary number. By connecting these branches to either 5 V or GND you can change the digits' values between 1 and 0. What do you expect to see on the DSO when changing their values? Do you already have an idea what the purpose of R_3, R_4, R_5 and R_6 is?

DA conversion The purpose of DA conversion is to produce a continuous analog signal out of the discrete digital one such that the produced signal approximates the original one as well as possible. The input signal will be a 3bit binary number. We can imagine a binary number as some code for a number value in binary position notation. This means that every position of the number has a certain value and will thus act multiplicative on the digit. In the case of binary numbers the first position has value 1, the second position has value 2, the third position has value 4, ..., the n-th position has value 2^{n-1}. So the actual value of a n-bit binary number b can be written as

$$b = \sum_{i=1}^{n} \text{<i-th digit>} \cdot 2^{i-1}. \tag{3}$$

We can try to naively apply this way of thinking to our problem: Every input bit can be weighted according to the value of the corresponding position in the binary system, and then all bits are summed together. For the purpose of weighting we use OpAmps with different amplification factors. The amplification factors are created by different resistances for each input bit. Since the value of a position of a binary number increases with a factor of 2 we need the same relation between the amplification factors of the OpAmps. From the physics lab 1 and 2 we know that the amplification factor V_{eff} of an invertin amplifier can be calculated with

$$V_{\text{eff}} = -\frac{R_{\text{feedback}}}{R_{\text{weight}}} \tag{4}$$

where R_{feedback} is the feedback resistance of the OpAmp and R_{weight} is the weighting resistance that we have to choose seperately for each bit.

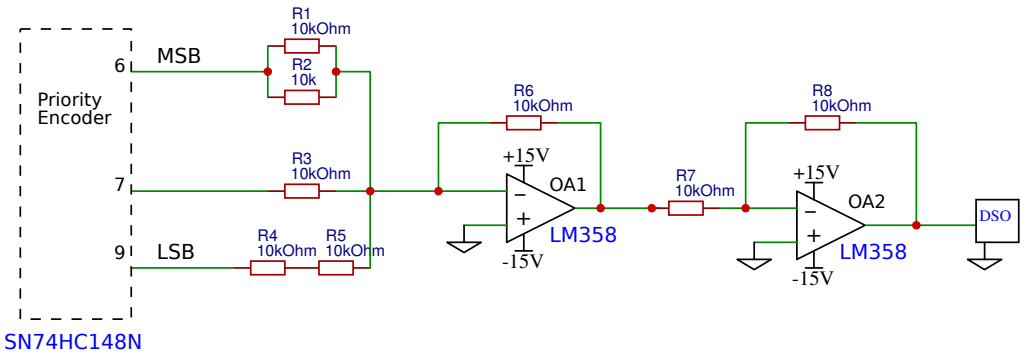

Figure 8: The 3bit DA converter circuit. From left to right there are the three inputs that may be connected directly to the priority encoder's outputs. The resistances R_1 to R_5 weight the inputs according to their significance. The first OpAmp OA_1 amplifies and inverts the signal while the second one OA_2 inverts it back without amplifying it again. Finally we connect the output to the DSO.

Exercise 5

Calculate the resistances $R_i, i = 1, 2, 3, 4, 5$ from Figure 8. First, identify the feedback resistance of the OpAmp. Start with an amplification factor of $V_{\text{eff}}^{\text{LSB}} = -\frac{1}{2}$ for the least significant bit (LSB).

One could naively assume that we can apply this method to any number n of bits just by doubling the resistance for each subsequent bit. This is unfortunately not practicable. Consider an 8bit DA convetrter that uses the same method as our circuit. The value of the 8th position is $2^7 = 128$. So if we choose to weight the MSB with $1\,\text{k}\Omega$ we must weigh the LSB with $128\,\text{k}\Omega$. But a resistance is typically fabricated with a tolerance of only 1%, which is in the case of the LSB already $(128.00 \pm 1.28)\,\text{k}\Omega$. So to uncertainty is already larger than the weight of the MSB! Because of this one uses a resistor ladder instead of our naiv implementation above for 6 and more bits. They use reiteratet units of resistances of values R and $2R$ for each bit. Commercial DA converters use this kind of conversion method. If you want to read more about this, check [11] and [1] for the keyword 'resistor ladder' (german: Leiternetzwerk oder R2R-Netzwerk).

3 Working with the converter

Exercise 6: Implementation

Set up the AD converter and the DA converter on two seperate breadboards as shown in Figure 7 and Figure 8. You will have to think first about how to arrange the three voltage channels of the power supply to provide all the desired voltages. (Hint: Use parallel and serial wiring. Also, all GND connection points are on the same level.) The entire setup should look similar to as shown in the Figures 9 and 10.

Exercise 7

Check the functionality of your converter by supplying a square signal, a sine signal and a triangle signal. In Figure 11 you can see a possible result for the three signals. Also get yourself a 4-Channel DSO from the Help Desk. Use one channel to display the input signal and the other three channels for the three output bits of the AD converter. By increasing the frequency of the input signal, find out how fast the AD converter works.

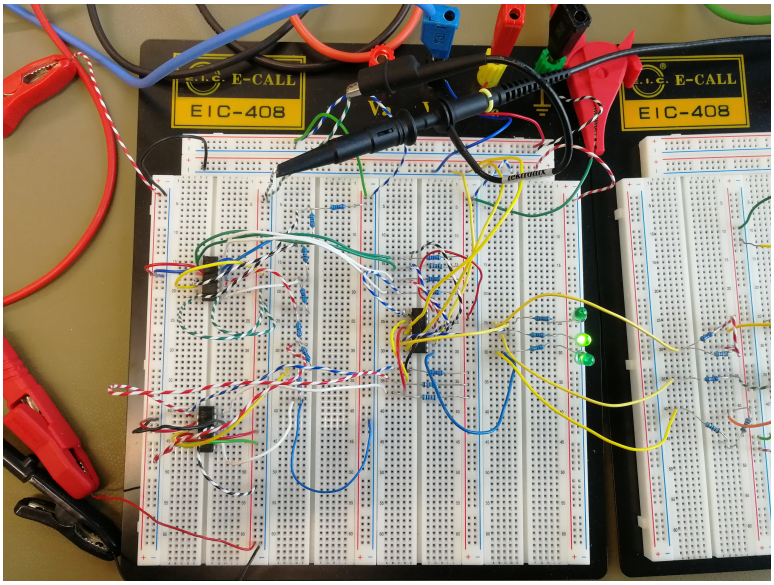

Figure 9: Picture of the real AD converter setup as implemented in the lab. The two ICs on the left hand side are the eight comparators with open collector output. The resistances in the third column of the breadboard form the linear voltage splitter. The IC on the right hand side is the priority encoder and the resistances around it are the pull-up resistances connected to the open collector output of the comparators. In the last column there are the three LEDs and their corresponding pre-resistances. The second breadboard shows a part of the connected DA converter.

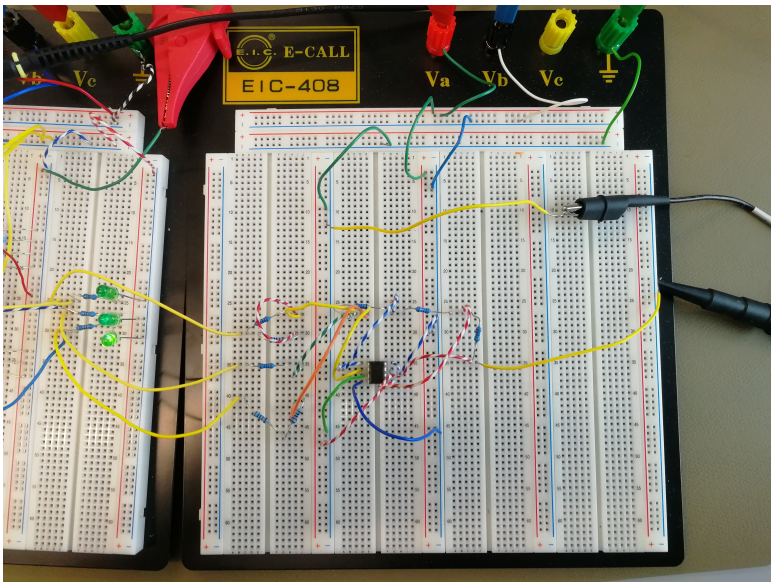

Figure 10: Picture of the real DA converter setup as implemented in the lab. From left to right there are the long yellow cables that connect the DA converter to the AD converter. In the first two columns there are the resistances that weight the bits according to their significance. The IC is home to the two OpAmps. The output of the IC is finally connected to the DSO.

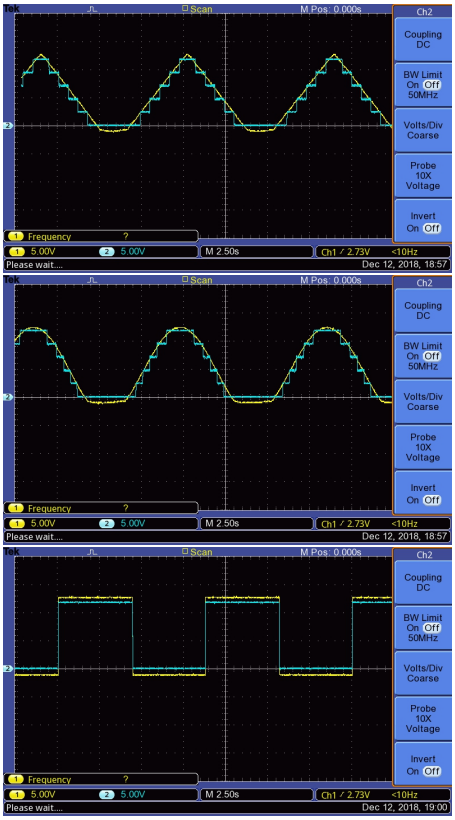

Figure 11: Screenshot of the DSO showing the original triangle/sine/square input signal (yellow) and the twice converted output signal (step function, blue). The input signal frequency is smaller than 1 Hz.

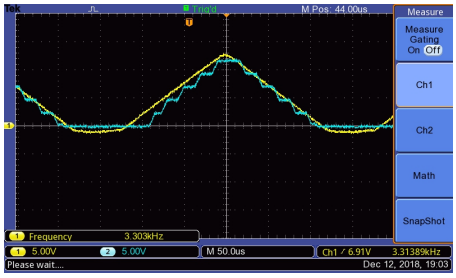

Figure 12: Screenshot of the DSO showing the original triangle input signal (yellow) at around 3.3 kHz and the twice converted output signal (step function, blue). Note the delay between the signals.

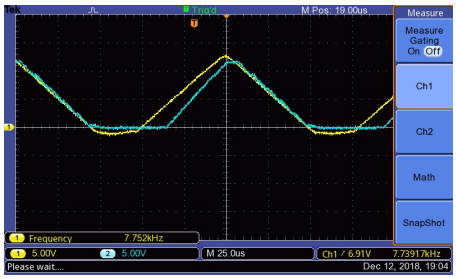

Figure 13: Screenshot of the DSO showing the original triangle input signal (yellow) at around 7.7 kHz and the twice converted output signal (step function, blue). Note that the converted signal does not have any distinguishable steps anymore.

Exercise 8

Increase the frequency and find an upper limit such that there is a clear delay between input and output of the DA(!) converter. Is there a speed difference between the whole setup and the AD converter alone? Compare this delay to the propagation delay of the gates. Do they coincide? If not, can you think of reasons what else causes a delay? Figure 12 gives you an idea of what you are looking for.

Exercise 9

As you increase the frequency of the signal more and more, you will find out that at some point the converted signal will not only be delayed but also appear without distinguishable steps. Also, you already know from the physics lab 1 and 2 that there is a frequency dependence of the OpAmp. Try to link those two statements. You could try to replace the OpAmp by a better one that amplifies still for higher frequencies to get an idea of the link. Use also the information you get from the data sheets. If you are not able to link those two phenomena try to think of another reason for the nondistinguishabilty of the steps for high frequencies. Figure 13 gives you an idea of what you are looking for.

4 Further ideas for the interested student

The following tasks are not obligatory. Nevertheless you are encouraged to develop these ideas further, if time permits.

Exercise 10

Modify the voltage splitter in the setup of the AD converter such that all discretization steps have size 1 V for a maximum input signal voltage of 7 V and are still well-defined. Note that you have only a limited amount of voltage sources available to implement your idea, so you will have to combine certain resistances in a clever way.

Exercise 11: A 4bit converter

Extend the circuit to a 4bit converter. How many comparators do you need now? Have a look into the data sheet of the priority encoder. There, it is explained how to wire up two encoder ICs to get higher bit resolution. For the DA converter try to apply the same way of thinking as in the exercises in section 4 but for 4 input bits.

References

[1] Tietze, U. und Schenk, C. Halbleiterschaltungstechnik, 5. Auflage, 1990
24. DA- und AD-Wandler, p. 654, Fig. 24.26
19.1.1 1-aus-n-Code, p. 457-458

[2] Datasheet LM339, Texas Instruments http://www.ti.com/lit/ds/symlink/lm339.pdf

[3] Datasheet SN74HC148, Texas Instruments http://www.ti.com/lit/ds/symlink/sn74hc148.pdf

[4] Datasheet LM358, Texas Instruments http://www.ti.com/lit/ds/symlink/lm358.pdf

[5] Wikipedia, Analog-Digital-Umsetzer, Realisierungsverfahren https://de.wikipedia.org/wiki/Analog-Digital-Umsetzer# Realisierungsverfahren

[6] Wikipedia, Functional diagram of single-slope DAC, February 2006 Translated from German to English Page URL: https://commons.wikimedia.org/wiki/File:Single_slope_dac.png File URL: https://upload.wikimedia.org/wikipedia/commons/3/35/Single_slope_dac.png Attribution: Dirkhb [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/)], via Wikimedia Commons

[7] Wikipedia, Analog-Digital-Umsetzer mit rückgekoppeltem DAU (Serieller Umsetzer), 6 March 2011 Translated from German to English Page URL: https://commons.wikimedia.org/wiki/File:DMT_ADU_mit_DAU.svg File URL: https://upload.wikimedia.org/wikipedia/commons/a/a8/DMT_ADU_mit_DAU.svg Attribution: Saure [CC BY-SA 3.0 (https://creativecommons.org/licenses/by-sa/3.0)], from Wikimedia Commons

[8] Wikipedia, Nyquist-Shannon-Abtasttheorem https://de.wikipedia.org/wiki/Nyquist-Shannon-Abtasttheorem

[9] Wikipedia, Komparator (Analogtechnik) https://de.wikipedia.org/wiki/Komparator_(Analogtechnik)

[10] Physics Lab 1+2 Manual, 37A, Analoge Elektronik https://ap.phys.ethz.ch/Anleitungen/37AAP.pdf

[11] https://en.wikipedia.org/wiki/Resistor_ladder