

Chapter 1

ADC (Analog to Digital Conversion)

Analog: Information or data that can be any real value at any instant of time. Physical world signals are of analog nature. For instance, a sinusoidal voltage or current. The voltage $V(t) = 5\sin(\omega t)$ is an analog voltage, that can take any value between $[-5, 5]$.

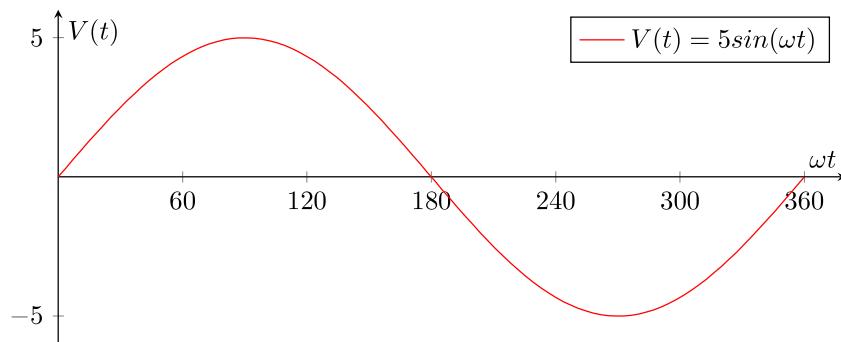


Figure 1.1: An analog signal

Digital: Information in the form of binary bit streams (0 & 1). Only *two* real values are used to represent the two binary bits. Computers and all digital devices need digital signals to operate. In most of the cases, a higher voltage is used to represent '1', while a lower voltage represents '0'.

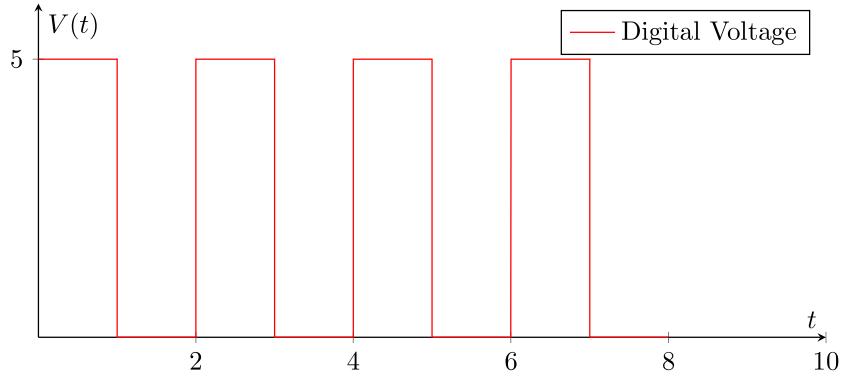


Figure 1.2: A digital signal

ADC is the process of converting an analog signal to a digital signal. The full process can be split into a few steps, and it can be modeled as the following flow chart:

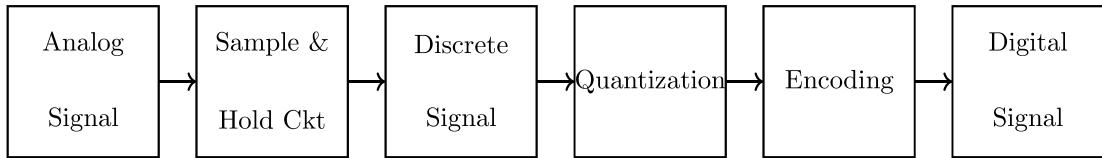


Figure 1.3: A flow chart of ADC process

Sample & Hold Circuit: The analog signal is sampled at regular intervals of a clock pulse. A *capacitor* is used to hold the peak value of the preceding interval until the next one arrives. Thus, instead of all the real values of the original signal, we get a signal with **reduced number of values** (still possibly greater than two), i.e., a *discrete signal*.

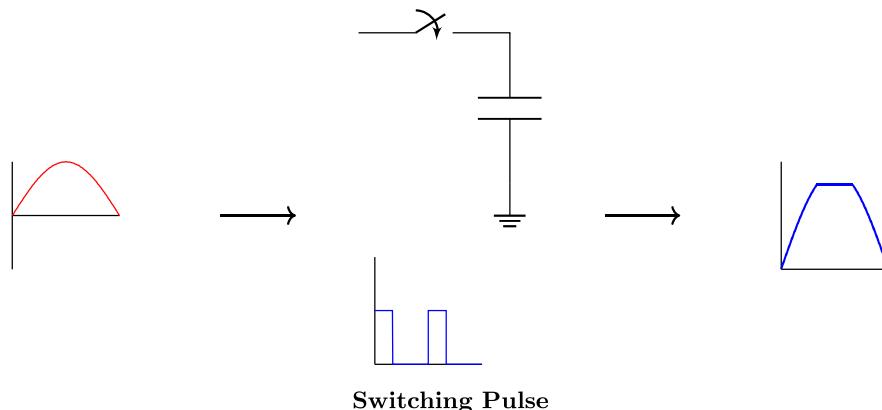


Figure 1.4: Sample & Hold circuit

Here, when the switch is closed, the capacitor charges according to the input analog signal. When the switch is open, it holds the last charged value until the next closing of the switch. This results in the output to miss out on the values of the input signal, that were between two successive closing of the switch. The faster the switching, the shorter is the duration of the pulse, and at very high frequencies, we effectively get about a single value of the input across the capacitor in each switching interval. These are called the *sampled values* of the input. These are then quantized using a certain quantization scheme.

Quantization: The sampled values from the previous stage still have a lot of values. The maximum allowable number of different values is determined by the *resolution* of the ADC process, which is the number of bits used in the output digital signal. We know that there be 2^n numbers for n bits. Thus, the maximum number of allowable values at the output is 2^n . We call these values the *quantization levels*. Now, the question is, which values do we select as the quantization levels? This depends on the *quantization scheme* we use. We will discuss two popular such methods next.

Mid-rise Quantization: In this method, the range of the input signal is divided into a set of *equal-sized intervals*, and the output levels are assigned to the *midpoints* of these intervals. The quantization levels are *symmetric* about the *signal zero*, but **do not include** the signal zero as a quantization level. Here, by signal zero we mean the midpoint between the signal maximum & minimum.

For example, consider a signal with maximum value of 5 and minimum value of -3. If we use 3 bits as the output digital signal, we will get $2^3 = 8$ quantization levels. Since the levels are equally spaced, the difference between any two successive quantization levels can be calculated as follows,

$$\text{Step size, } \Delta = \frac{V_{max} - V_{min}}{2^n} \quad (1.1)$$

Where, V_{max} & V_{min} are the maximum and minimum values of the input analog signal respectively. For our example, equation 1.1 gives,

$$\Delta = \frac{5 - (-3)}{2^3} = 1$$

Thus, the quantization intervals will be as the following, starting at signal minimum:

Quantization Interval	Quantization Level
[-3, -2)	-2.5
[-2, -1)	-1.5
[-1, 0)	-0.5
[0, 1)	0.5
[1, 2)	1.5
[2, 3)	2.5
[3, 4)	3.5
[4, 5]	4.5

Table 1.1: Mid-rise Quantization

As you can see, there is no 1 among the quantization levels, which is the signal zero for our example.

You might wonder, the sampled signal from the sample & hold circuit still has more values than the number of the quantization levels. What do we do with the values other than the quantization levels? Do we discard them? No. We map all the sampled values according to table 1.1. A sampled value will be changed to the quantization level value of the quantization interval it lies within. For instance, suppose a sampled value is -1.2. It falls within the quantization interval [-2,-1], for which the quantization level is -1.5. Thus, -1.2 will be treated as -1.5 after the quantization process.

Mid-tread Quantization: The main difference of this scheme with mid-rise quantization is that, it *uses* the signal zero as a quantization level. Consequently, we cannot make all the intervals of equal length, even if we wanted to. Why? Well if you closely observe table 1.1, you will realize that it is not possible to include the signal zero as the quantization level if we begin and end at signal minimum and maximum respectively, and also keep all the intervals of equal length. Thus, we have to adjust at the two ends. A mid-tread quantization scheme for our previous example signal with unequal intervals at the first & last one is shown below:

Quantization Interval	Quantization Level
[-3, -2.5)	-2.75
[-2.5, -1.5)	-2
[-1.5, -0.5)	-1
[-0.5, 0.5)	0
[0.5, 1.5)	1
[1.5, 2.5)	2
[2.5, 3.5)	3
[3.5, 5]	4.25

Table 1.2: Mid-tread Quantization

Encoding: Now that we have the quantized signal with only 2^n different values, we need to convert them to binary bit streams of n bits to get our final digital output. This process is called *encoding*. In this step, the quantized signal is assigned a bit stream each. Simply, the 2^n quantization levels are represented with the 2^n binary numbers possible with n bits. For our previous example, the encoding would be as follows:

Quantization Level	Encoded Output
-2.5	000
-1.5	001
-0.5	010
0.5	011
1.5	100
2.5	101
3.5	110
4.5	111

Table 1.3: Encoding of Mid-rise Quantized Signal

Now we shall go the main part. Our main focus in this course is how this encoding is done. For this we will discuss two ADC circuits that do the job — Flash & Dual Slope ADC.

Flash ADC

The Flash ADC circuit for a 3-bit ADC is given below:

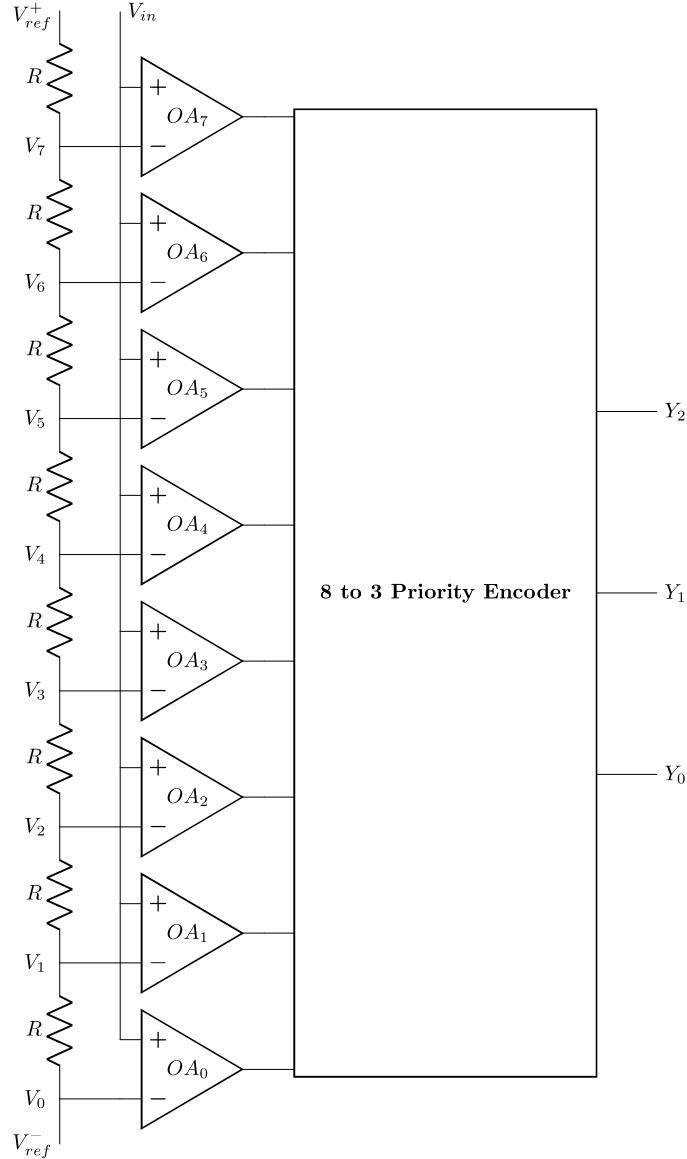


Figure 1.5: 3-bit Flash ADC (Mid-rise Quantization)

Let's discuss its several parts one by one.

Op Amps: Try to recall your learning on Op Amps from CSE251. When they are used in *open-loop* configuration, they act as comparators. The output from them is the high bias voltage V_H if $V_+ > V_-$. Otherwise, the output is the low bias voltage, V_L .

As we can see from figure 1.5, the non-inverting or positive terminal takes in the analog input voltage, while the inverting or negative terminal is connected to a resistance ladder in between two resistances. We should recall also that, in any configuration, ideal Op Amps do not draw any current in either of their terminals. Hence, these connections from the inverting terminals to the resistance ladder, are just taking the voltage, and are effectively open circuits. Consequently, the resistances are actually in series.

Priority Encoder: A Priority Encoder is a special type of encoder, that outputs the binary of the serial number of the *highest* input terminal that receives high voltage. It ignores the input terminals with a lower serial number than the one mentioned. The truth table of an 8 to 3 Priority Encoder is the following:

I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	\times	0	0	1
0	0	0	0	0	1	\times	\times	0	1	0
0	0	0	0	1	\times	\times	\times	0	1	1
0	0	0	1	\times	\times	\times	\times	1	0	0
0	0	1	\times	\times	\times	\times	\times	1	0	1
0	1	\times	\times	\times	\times	\times	\times	1	1	0
1	\times	1	1	1						

Table 1.4: Truth table of an 8-to-3 Priority Encoder

Here, ‘ \times ’ represents ‘don’t care’ logic.

Analysis of Operation: As we stated earlier, the resistances in the ladder are in series. Thus, the total resistance of the ladder is $8R$. Using Ohm’s law, we can write the current in this resistance ladder as,

$$I = \frac{V_{ref}^+ - V_{ref}^-}{8R} \Rightarrow IR = \frac{V_{ref}^+ - V_{ref}^-}{8}$$

Here, 8 resistances are used, because the ADC is a 3-bit one. Thus, the *resistance count* of a Flash ADC is 2^n . Let’s bring back our previous example of the analog signal with maximum value 5 and minimum value -3. If we use these values as the reference voltages of the resistance ladder, then we get,

$$IR = \frac{5 - (-3)}{8} = 1$$

Which is the step size, Δ . Now, using KVL along the ladder we can write,

$$V_1 = V_0 + IR = -3 + 1 = -2$$

This was the lower boundary of the 2nd quantization interval of mid-rise quantization, [-2,-1). In general, we can observe that,

$$V_n = V_0 + n \times IR = V_0 + n \times \Delta$$

The whole circuit operates this way — the input voltage is compared with the inverting terminal voltages of all the Op Amps (V_n). The Op Amps for which $V_{in} > V_n$, the output is high. Thus the corresponding input of the encoder, I_n is high. The other Op Amps will output low voltage

and the corresponding encoder inputs will be low. Now, the encoder output is decided based on what is the highest serial of the Op Amps giving high output, as explained in table 1.4. For our example, suppose the input voltage is -1.2 V. This is higher than the V_- of OA_0 & OA_1 (-3 V & -2 V respectively). Thus, I_0 & I_1 will be high and other inputs low. From table 1.4 we see that the output of the encoder for this case is $(001)_2$. If you recall from the discussion on encoding, this is the desired output for the mentioned input voltage. The circuit shown in figure 1.5 is however, not the only possible Flash ADC. Depending on the quantization scheme used, it may be modified. For instance, if we had used mid-tread quantization instead for our example, the circuit would look like this:

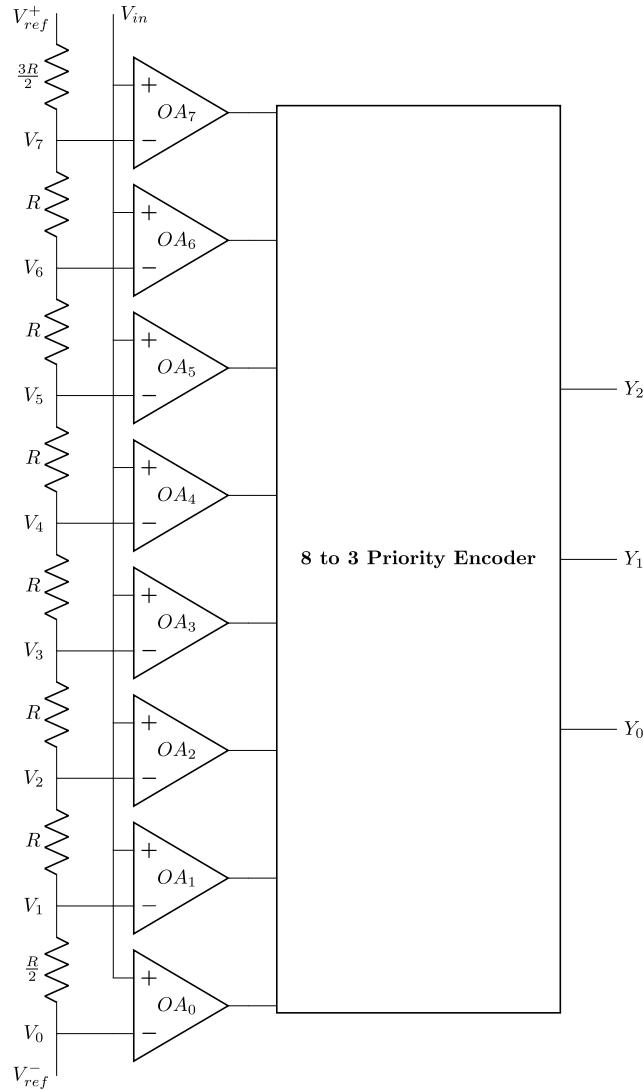


Figure 1.6: 3-bit Flash ADC (Mid-tread Quantization)

Through a similar analysis as in the mid-rise case on figure 1.6 should lead you to concluding that the inverting terminal voltages are again equal to the lower boundaries of the quantization

intervals of table 1.2.

Some interesting facts to note here:

1. **Redundancy of the 0th Op Amp:** The input signal is always greater than or equal to its minimum value, which corresponds to the V_- of the 0th Op Amp. Therefore, it will always output high. Instead of using OA_0 , we can directly apply a high voltage at I_0 of the encoder. Hence, the *Op Amp count* in a Flash ADC is either 2^n or $2^n - 1$.
2. **Why the name ‘Flash’ ADC?** Because its *conversion speed is very high*—it can convert a sampled value in a *single clock cycle*.
3. **Disadvantage:** A Flash ADC uses a large number of components (resistors and Op Amps), which makes it costly and power-hungry.

Dual Slope ADC

To overcome the disadvantages of a Flash ADC, we introduce Dual Slope ADC, a cheaper but slow alternative. A generalized diagram of the Dual Slope ADC is given below:

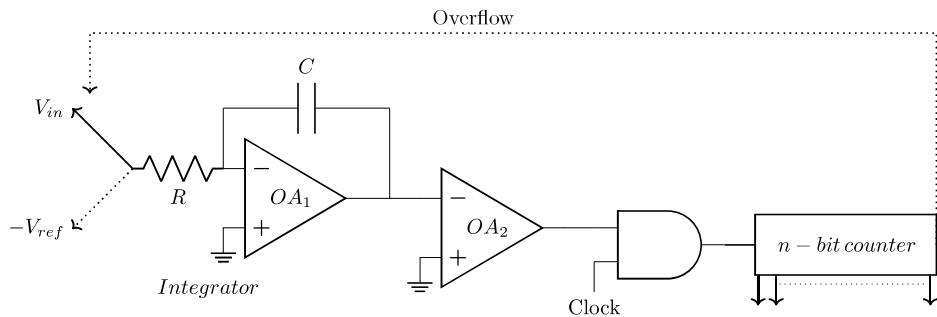


Figure 1.7: Dual Slope ADC

Let's discuss the components one by one.

Integrator: OA_1 , R & C form an integrator, input to which switches between $-V_{ref}$ & V_{in} . From CSE251, we should know that if the input voltage to the integrator is V_i , then the output is,

$$V_o(t) = -\frac{1}{RC} \int_0^t V_i dt + V_o(0) \quad (1.2)$$

Comparator: OA_2 is a simple comparator to which, the output from the integrator is connected, at its inverting terminal. The non-inverting terminal is grounded. Thus, it will output high, when the integrator's output is negative.

AND gate & Clock: The output from the comparator is fed to an AND gate along with a clock signal of high frequency. The AND gate will give a high output if both the clock pulse and the comparator's output are high.

n-bit counter: An n-bit counter counts from 0 to 2^n-1 , i.e. 2^n times, for each high input. The output from this counter is the binary of the current serial (between 0 to 2^n-1) it has counted up to. For example, consider a 3-bit counter. It counts from 0 to 7. So, its output will change from $(000)_2$ to $(111)_2$ by one step at each high input to the counter. Thus, if we say the counter has counted 4 times since the start, it has actually counted from 0 up to 3, and its current output will be $(011)_2$. Once it reaches 2^n-1 , the next high input causes it to overflow, and the counter resets to 0.

Analysis of Operation: We will take a step by step approach to understand the function of the Dual Slope ADC.

1. **Integration:** We discussed earlier that the input to the integrator switches between input & a reference voltage. In each separate conversion of a sampled analog input, the input voltage is connected to the integrator input for a fixed amount of time t_1 . As we said that we use high sampling frequencies, which effectively makes the sampled values a single constant voltage. We can observe from equation 1.2, that a constant input voltage would cause the output of the integrator to be as follows(assuming 0 initial voltage):

$$V_o(t) = -\frac{1}{RC} \times V_{in} \times t$$

This is a straight line equation with respect to time, for which the slope is $-\frac{V_{in}}{RC}$. Let the voltage output from the integrator, after t_1 is V_1 . Then we have,

$$V_1 = -\frac{V_{in}t_1}{RC} \quad (1.3)$$

The time t_1 depends on the number of bits used in the output digital signal, and on the frequency of the clock pulse in the following manner,

$$t_1 = \frac{2^n}{f_{CLK}} \quad (1.4)$$

The time t_1 is actually the time taken for the n-bit counter to count 2^n times starting from 0 to $2^n - 1$. Its expression can be easily driven by unitary method. As we said the counter counts for every high clock pulse, number of pulses thus needed for counting 2^n times is also 2^n . If the clock frequency is f_{CLK} , time taken for 2^n pulses = $2^n \times f_{CLK}$. This process is showed in the following graph:

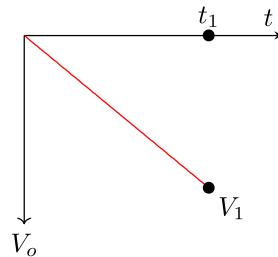


Figure 1.8: Integrator output after t_1

2. **De-integration:** Next, the switch at the input of the integrator connects to the negative reference voltage $-V_{ref}$. This is also a constant voltage. Thus, we will again get a straight

line output from the integrator, but this time the slope will be positive. Also, the initial output voltage is not 0 this time, rather, V_1 . This rising (positive slope) nature of the output causes the value to increase from V_1 to 0. The time taken for the output of the integrator to become 0 from V_1 is t_2 . Using equation 1.2 again, we can write for this case,

$$\begin{aligned}
 V_o(t_1 + t_2) &= -\frac{V_{ref}}{RC} \int_{t_1}^{t_1+t_2} dt + V_o(t_1) \\
 &\Rightarrow 0 = \frac{V_{ref}}{RC} (t_2 + t_1 - t_1) + V_1 \\
 &\Rightarrow \frac{V_{ref} t_2}{RC} - \frac{V_{in} t_1}{RC} = 0 \\
 &\Rightarrow t_2 = \frac{V_{in} t_1}{V_{ref}} \\
 &\Rightarrow t_2 = \frac{2^n}{f_{CLK}} \cdot \frac{V_{in}}{V_{ref}}
 \end{aligned} \tag{1.5}$$

Since all other parameters in equation 1.5 are constants, t_2 is proportional to the input voltage V_{in} . Which means, the larger the sampled value, the longer is t_2 .

The output of the integrator, after this process would look like the following:

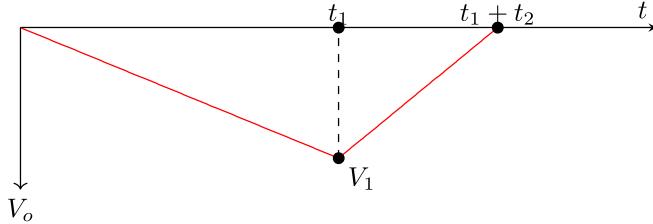


Figure 1.9: Integrator output after t_2

3. **Comparator AND Clock:** We see that the integrator output in figure 1.9 is always negative, during the whole conversion process ($t_1 + t_2$). Since this output is fed to the inverting terminal of the comparator (OA_2), the output of OA_2 will always be high. Consequently, the AND gate function reduces from Comparator AND Clock to Clock (1 AND A = A). This means, the AND gate will output high value whenever the clock pulse is high. Only after t_2 , when the integrator's output becomes 0, the comparator will give low output. Then the AND gate will block the clock pulse from the counter, marking an end to the counting.
4. **Counting of the counter:** The n-bit counter is effectively receiving the clock as its input. At each clock pulse, it counts one by one. During 0 to t_1 , it counts from 0, up to 2^n-1 . Then it overflows, and this causes the input of the integrator to switch from V_{in} to V_{ref} . Then the counter resets to 0, and begins counting for the 'De-integration' phase. The formula of t_1 we showed in equation 1.4, can be derived using unitary method with the help of this

concept.

The counter counts once every T_{CLK} time

The counter counts 2^n times in $T_{CLK} \times 2^n$ time

$$\therefore t_1 = 2^n \times T_{CLK} = \frac{2^n}{f_{CLK}}$$

Here, T_{CLK} is the period of the clock pulse.

As we saw back in equation 1.5, that t_2 is proportional to the input sampled signal, the counter counts up to a value proportional to this input voltage within t_2 ($t_2 \times f_{CLK} = \frac{V_{in}}{V_{ref}} \propto V_{in}$).

Hence, the output of the counter after t_2 is the binary of $N = (t_2 \times f_{CLK} - 1)$ (since the counter counts from 0 and $t_2 \times f_{CLK}$ times).

In a word, we get a digital output from the n-bit counter proportional to the input sampled signal. The bit count of the counter depends on the number of quantization levels we use. For 2^n quantization levels, we need an n-bit counter. For instance, if we want 8 quantization levels, we need a 3-bit counter. One thing to note here, is that, t_1 is the same for different sampled inputs. However, t_2 varies. Thus, we will have different V_1 values after the same t_1 , while the conversion for each finishes at different instants ($t_1 + t_2$). All these are visualized below for multiple samples:

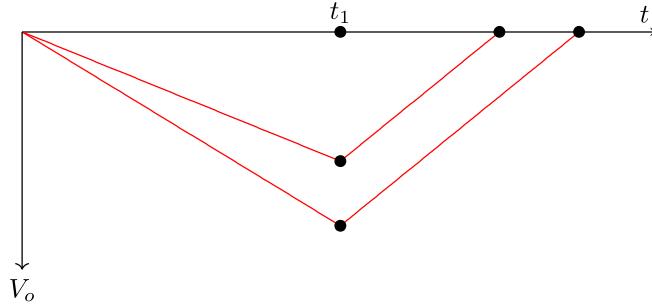


Figure 1.10: Integrator output for different sampled inputs

Let's discuss some important facts regarding Dual Slope ADCs.

1. Let's consider the decimal of the output, N. As we said, $N = (t_2 \times f_{CLK}) - 1$. From which we can write, $t_2 = \frac{N+1}{f_{CLK}}$. Comparing this with equation 1.5,

$$V_{in} = \frac{N+1}{2^n} \cdot V_{ref} \quad (1.6)$$

2. This is not the only possible Dual Slope ADC circuit. There is a common modification to this—the input voltage is applied in opposite polarity, and the reference voltage used is then positive. This then causes the *integration* phase to have a positive slope, and the *de-integration* phase takes a negative slope. However, the analysis and other formulas remain the same. Just, the whole output from the integrator is mirrored with respect to x-axis. The integrator's output is then as shown:

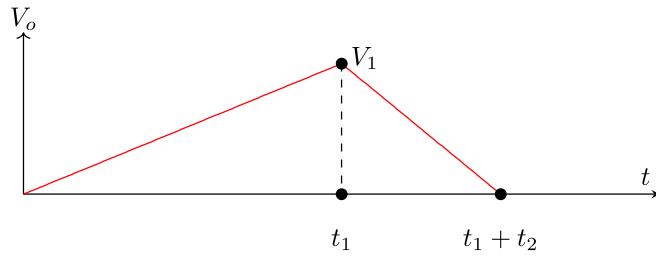


Figure 1.11: Integrator output if V_{in} & V_{ref} are reversed

3. Well we haven't discussed the context behind the name 'Dual Slope' yet, but you should have guessed it by now. As we see in both figures 1.9 & 1.11, the slope of the integration, and de-integration phases are different. First one depends on the value of the sampled input, while the second one is the same for all samples, and depends on the reference voltage instead.
4. The number of *components* used are significantly lower than the equivalent Flash ADC circuit. For instance, a 4-bit Flash ADC would require at least 2^4 or 16 resistors, and $2^4 - 1$ or 15 Op Amps. Whereas, the Dual Slope ADC needs two Op Amps and only one resistor & capacitor irrespective of the resolution or bit count.
5. Dual Slope ADCs are really slow, and take a lot of time for conversion compared to Flash ADCs.

Problems Roster

Problem 1: Design a 3-bit Flash ADC for a signal with maximum & minimum value of 10 V & 0 V respectively. The quantized levels of the ADC should be uniform (equal step-size). Answer the following:

(a)	Calculate the total number of resistors & Op Amps required.
(b)	Calculate the 1LSB value or step size.
(c)	Find the quantization levels & plot the D_{out} (digital output) vs V_{in} plot.
(d)	If the input is 7 V: (i) Find the quantization interval the input lies in. (ii) What is the digital output for this input?
(e)	If the digital output is 110_2 , what are the maximum & minimum values of input that would produce this result?
(f)	If an additional bit was desired at the output, what would be the required number of components?

Solution:

$$(a) N_R = 2^n = 2^3 = 8$$

$$N_{OA} = 2^n \text{ or } 2^n - 1 = 8 \text{ or } 7$$

$$(b) V_{1LSB} = \Delta = \frac{V_{max} - V_{min}}{2^n} = \frac{10 - 0}{8} = 1.25 \text{ V}$$

(c) The quantization levels are as follows:

Quantization Interval	Quantization Level
0-1.25	0.625
1.25-2.5	1.875
2.5-3.75	3.125
3.75-5	4.375
5-6.25	5.625
6.25-7.5	6.875
7.5-8.75	8.125
8.75-10	9.375

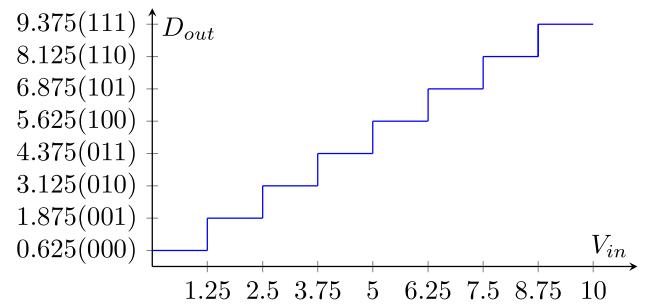


Figure: D_{out} vs V_{in} graph

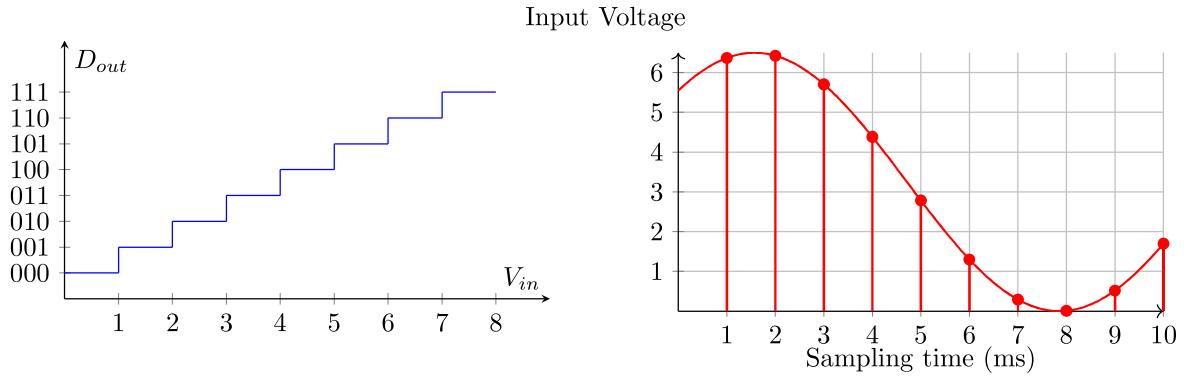
(d)

(i) 6.25-7.5

(ii) 101

(e) Maximum = 8.25 V & minimum = 7.5 V.

(f) For 4-bit Flash ADC, $N_{OPAMP} = 2^4$ or $2^4 - 1$. $N_R = 2^4$.

Problem 2:

- | | |
|-----|---|
| (a) | Design a 3-bit Flash ADC for the given input-output characteristics. |
| (b) | An unknown signal is passed to the ADC as input. The ADC takes samples at a rate of 1 kHz. In the graph, the input signal is shown where the x-axis represents the time, and the y-axis represents voltage at any specific time. The sampling instances are given in the figure. Find the encoded output that represents the analog signal inside the given time frame. |

Solution:

(a)

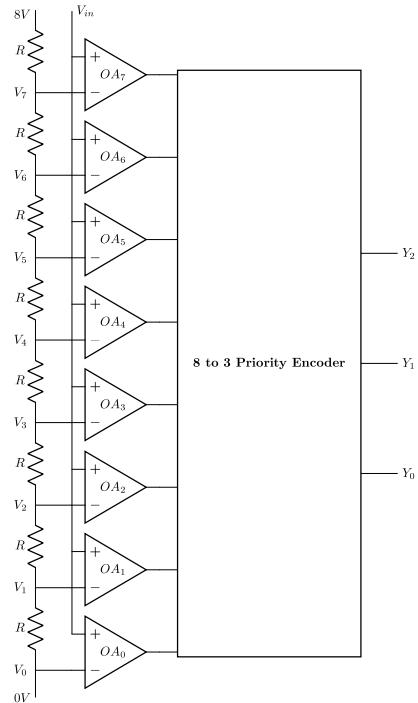


Figure 1.12: 3-bit Flash ADC for Problem 2(a)

(b)

Sampling Instances	Input Voltage	Quantization Interval	Encoded Value
1 ms	6.3 V	6 V - 7 V	110
2 ms	6.4 V	6 V - 7 V	110
3 ms	5.7 V	5 V - 6 V	101
4 ms	4.4 V	4 V - 5 V	100
5 ms	2.8 V	2 V - 3 V	010
6 ms	1.3 V	1 V - 2 V	001
7 ms	0.3 V	0 V - 1 V	000
8 ms	0 V	0 V - 1 V	000
9 ms	0.5 V	0 V - 1 V	000
10 ms	1.7 V	1 V - 2 V	001

Problem 3:

A Dual Slope ADC has $V_{ref} = 5$ V, and an 8 bit counter that outputs an 8-bit representation of the input signal. A 1 MHz clock is used for the clock and counter circuits.

- | | |
|-----|--|
| (a) | Determine the input voltage range, number of steps and step size (resolution) of the ADC. |
| (b) | For a specific input voltage, the ADC outputs a count of m=100. Determine the input voltage. |
| (c) | Determine the total time required to get the reading in (b). |
| (d) | Calculate the maximum sampling rate of the ADC. |
| (e) | A person wishes to get a sampling rate 4 times that of the current circuit. Determine how many bits should be used for the counter without changing any other system parameters. |

Solution:

(a) Input voltage range: 0 V - 5 V

Number of steps = $2^8 = 256$

$$\text{Step size, } \Delta = \frac{V_{ref}}{2^n} = \frac{5}{256} = 0.0195 \text{ V}$$

(b) We know from equation 1.6, $V_{in} = \frac{N+1}{2^n} V_{ref}$

$$\therefore V_{in} = \frac{100 + 1}{256} \times 5 = 1.973 \text{ V}$$

(c) We know from equation 1.4, $t_1 = \frac{2^n}{f_{CLK}}$

$$\therefore t_1 = \frac{2^8}{1 \times 10^6} \text{ s} = 0.256 \text{ ms}$$

From equation 1.5, $t_2 = \frac{2^n}{f_{CLK}} \cdot \frac{V_{in}}{V_{ref}}$

$$\therefore t_2 = \frac{2^8}{1 \times 10^6} \cdot \frac{1.973}{5} \text{ s} = 0.101 \text{ ms}$$

Therefore, total conversion time, $t = t_1 + t_2 = 0.256 + 0.101 = 0.357 \text{ ms}$

(d) Maximum conversion time is limited by the time required for worst case conversion of input $= V_{ref} = 5 \text{ V}$. For this case $t_1 = t_2 = 0.256 \text{ ms}$.

Thus, worst case conversion time, $t_{max} = 2 \times t_1 = 2 \times 0.256 = 0.512 \text{ ms}$.

$$\therefore \text{Maximum sampling rate} = \frac{1}{t_{max}} = \frac{1}{0.512 \times 10^{-3}} \text{ Hz} = 1953.125 \text{ Hz}$$

(e) From (d) we see, maximum sampling rate $\propto \frac{1}{2^n}$. Thus,

$$\frac{f_{samp_1}}{f_{samp_2}} = \frac{2^{n_2}}{2^{n_1}} \Rightarrow \frac{2^{n_2}}{2^8} = \frac{1}{4} \Rightarrow n_2 = 6$$

Problem 4:

Suppose a sinusoidal signal with peak-to-peak value of 6 V & frequency of 1 kHz needs to be encoded using Dual Slope ADC. What changes are necessary to the Dual Slope ADC in *Problem 3* to get a successful conversion?

Solution: If we closely observe, there is another limitation of the Dual Slope ADC of figure 1.7—it cannot deal with bipolar input signals (i.e. signals with both positive & negative values).

The sinusoidal signal mentioned in the question ranges between -3 to 3. To handle this signal and to get a successful conversion using the Dual Slope ADC, we first need to make all of its values positive. This can be done by adding an *appropriate offset* to the sine wave. The goal is to get all of its values within the range of 0 to a maximum value, which is the requirement for our Dual Slope ADC.

So, for a sine wave of amplitude 3 V (peak-to-peak value of 6 V), we need to add 3 V of offset. This will lift all the values of the sine wave between 0 V to 6 V. This is illustrated in the following figure:

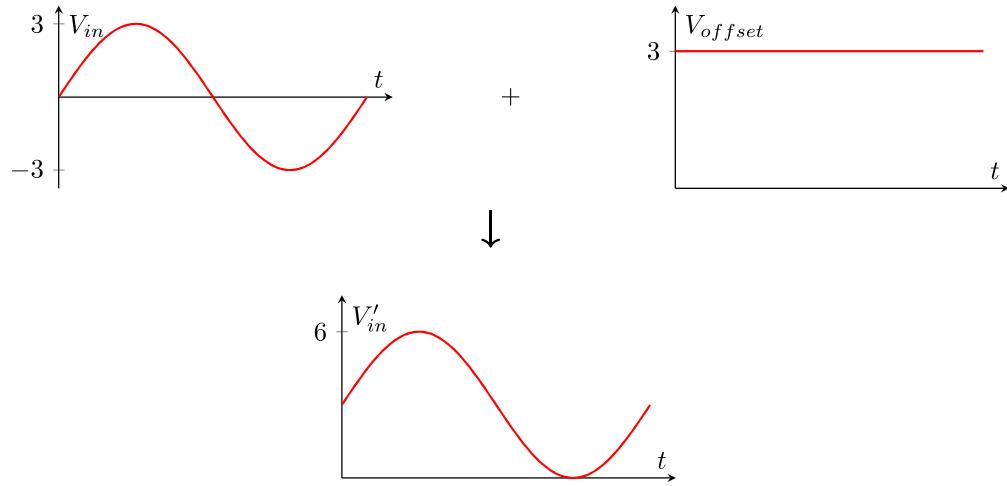


Figure 1.13: Adding offset to input signal

But the question is, how do we achieve this addition of offset? Well, it's easier than doing the ADC. Recall from CSE251, we studied inverting adders. If we simply apply the sinusoidal input

to one input of an inverting adder, and a DC voltage of -3 V to another, we can get V'_{in} as in figure 1.13. Thus the full circuit including this adder would be:

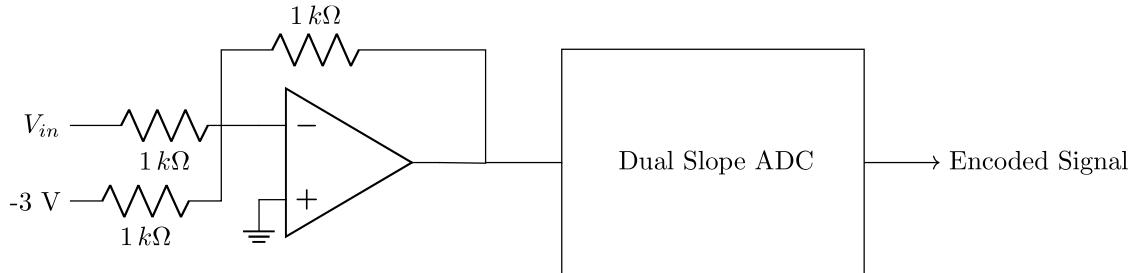


Figure 1.14: Modified Dual Slope ADC for handling bipolar input

Notice that we do not need to invert the input sinusoidal signal here, because it is bipolar. Two important jobs are still to be done. The reference voltage of the Dual Slope ADC also needs to be changed. As the input signal ranges between 0 V to 6 V now, the new reference voltage $V'_{ref} = 6$ V.

Also, we know that the minimum sampling frequency (Nyquist rate) must be twice of the input signal. So, we need to make sure that the maximum sampling rate of the Dual Slope ADC be at least 2 kHz. Thus, we need to modify the clock frequency.

$$\text{Worst conversion time, } t = 2 \times t_1 = 2 \times \frac{256}{f'_{CLK}}$$

$$\therefore f'_{CLK} = \frac{512}{t} = \frac{512}{\frac{1}{f_{Nyquist}}} = \frac{512}{\frac{1}{2 \times 10^3}} = 1.024 \text{ MHz}$$

Problem 5:

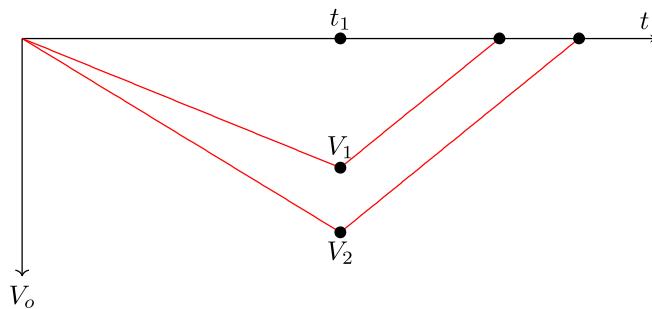


Figure 1.15: Integrator output vs time

For a 6-bit Dual Slope ADC $V_1 = -6$ V, $V_2 = -8$ V in figure 1.15 and the maximum allowable sampling frequency is 1.5625 kHz. The time constant τ of the integrator is 0.5 ms.

(a)	Find the two input voltages.
(b)	If the reference voltage is 15 V, find the digital outputs for the inputs found in (a).
(c)	Find the encoded output for which the slope of the integration phase of the integrator would be the average of the two given.

Solution:

(a)

$$V_{in_1} = V_1 \times \frac{RC}{t_1} = -(-6) \times \frac{0.5 \times 10^{-3}}{t_1}$$

Now, worst case conversion time, $2t_1 = \frac{1}{1.5625 \times 10^3} = 0.64 \text{ ms}$, $\therefore t_1 = 0.32 \text{ ms}$.

$$\therefore V_{in_1} = \frac{3 \times 10^{-3}}{0.32 \times 10^{-3}} = 9.375V$$

Similarly,

$$V_{in_2} = \frac{-(-8) \times 0.5 \times 10^{-3}}{0.32 \times 10^{-3}} = 12.5V$$

(b)

$$N_1 + 1 = \frac{V_{in_1}}{V_{ref}} \times 2^6 = \frac{9.375}{15} \times 64 = 40 \Rightarrow N_1 = 39$$

$$N_2 + 1 = \frac{V_{in_2}}{V_{ref}} \times 2^6 = \frac{12.5}{15} \times 64 = 53 \text{ (floored value)} \Rightarrow N_2 = 52$$

$$\therefore D_1 = \text{binary of } N_1 = 100111$$

$$\therefore D_2 = \text{binary of } N_2 = 110100$$

(c) Let slopes of integration for V_{in_1} & V_{in_2} are m_1 & m_2 respectively.

$$m_1 = -\frac{V_{in_1}}{RC} = -\frac{9.375}{0.5} = -18.75 \text{ V/ms}$$

$$m_2 = -\frac{V_{in_2}}{RC} = -\frac{12.5}{0.5} = -25 \text{ V/ms}$$

$$\therefore m = \frac{m_1 + m_2}{2} = -21.875 \text{ V/ms} \Rightarrow -\frac{V_{in}}{RC} = -21.875 \Rightarrow V_{in} = 0.5 \times 21.875 = 10.9375V$$

$$\therefore N + 1 = \frac{V_{in}}{V_{ref}} \times 2^6 = \frac{10.9375}{15} \times 64 = 46 \text{ (floored value)} \Rightarrow N = 45$$

$$\therefore D = \text{binary of } 45 = 101101$$

Problem 6:

If for the 3-bit Flash ADC shown in figure 1.6, $V_2 = -1 \text{ V}$ and $V_6 = 7 \text{ V}$ —

(a)	Find the reference voltages V_{ref}^+ & V_{ref}^- .
(b)	What value of offset needs to be added to the input signal, for which the Flash ADC in (a) is designed, so that it can be implemented using an 8-bit Dual Slope ADC with a 2.5 MHz clock? What will be the reference voltage of the Dual Slope ADC?
(c)	Find the average conversion time of the Dual Slope ADC mentioned in (b), for the highest quantization interval in (a).

Solution:

(a) We see in figure 1.6, $V_2 = V_{ref}^- + I \times 1.5R$. Also, $V_6 = V_{ref}^- + I \times 5.5R$.

$$V_{ref}^- + 1.5 \times IR = -1$$

$$V_{ref}^- + 5.5 \times IR = 7$$

Solving the above two equations, $V_{ref}^- = -4V$, $IR = 2V$.

$$V_{ref}^+ = V_{ref}^- + 0.5IR + 6IR + 1.5IR = -4 + 8IR = -4 + 8 \times 2 = 12V$$

(b) $V_{offset} = 4V$. $V_{ref} = (V_{ref}^- + V_{ref}^+) = (12 - (-4)) = 16V$

(c) Highest quantization interval in (a): 9 V - 12 V.

$$\text{Total conversion time for input of } 9V, t_{9V} = \frac{2^n}{f_{CLK}} + \frac{9 + V_{offset}}{V_{ref}} \times \frac{2^n}{f_{CLK}} = \frac{2^8}{2.5 \times 10^6} \left(1 + \frac{9+4}{16}\right) = 0.1856ms$$

$$\text{Total conversion time for input of } 12V, t_{12V} = \frac{2^n}{f_{CLK}} + \frac{12 + V_{offset}}{V_{ref}} \times \frac{2^n}{f_{CLK}} = \frac{2^8}{2.5 \times 10^6} \left(1 + \frac{12+4}{16}\right) = 0.2048ms$$

Thus, the average conversion time for this quantization interval, $t_{avg} = \frac{0.1856+0.2048}{2} = 0.1952ms$

