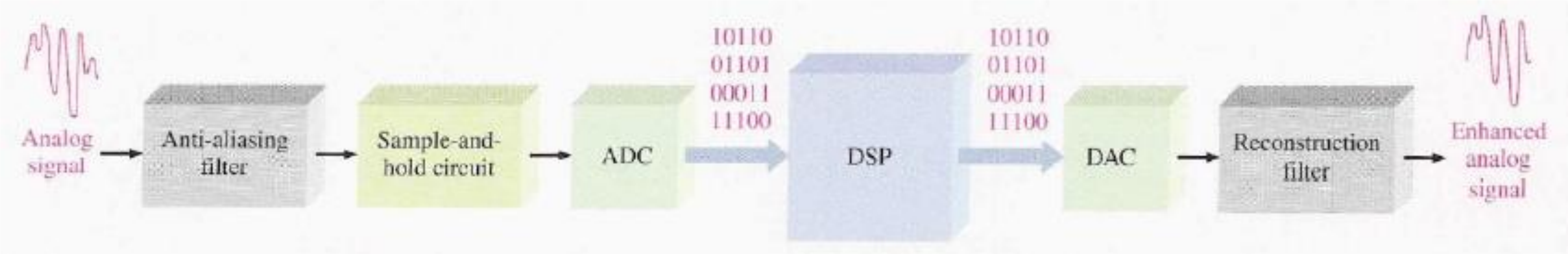
# Lecture -7
# Introduction to Digital Signal Processing 1

Prepared By: Asif Mahfuz

# Introduction

- Digital Signal Processing coverts signals that naturally occur in analog form, such as sound, video and information from sensors, etc. to digital form and uses digital techniques to enhance and modify analog signal data for various applications.



# Sampling and Filtering

- First the analog signal is passed through the Anti-aliasing filter (low pass filter) to eliminate harmonic frequencies above a certain specified frequency determined by the Nyquist frequency.
- Then the sampling and hold circuit performs two operation, the first of which is sampling.
- Sampling is the process of taking a sufficient number of discrete values at points on a waveform that will define the shape of the waveform.
- An analog signal can constitute signals of various frequencies.
- **Sampling Theorem** states that, in order to represent an analog signal, the sampling frequency , $\mathbf{f_{sample}}$, must be at least twice the highest frequency component $\mathbf{f_{a(max)}}$ of the analog signal.
- The frequency $\mathbf{f_{a(max)}}$ is known as the **Nyquist frequency** and is expressed in

$$\mathbf{f\_sample \geq 2f_{a(max)}}$$

# Sampling and Filtering

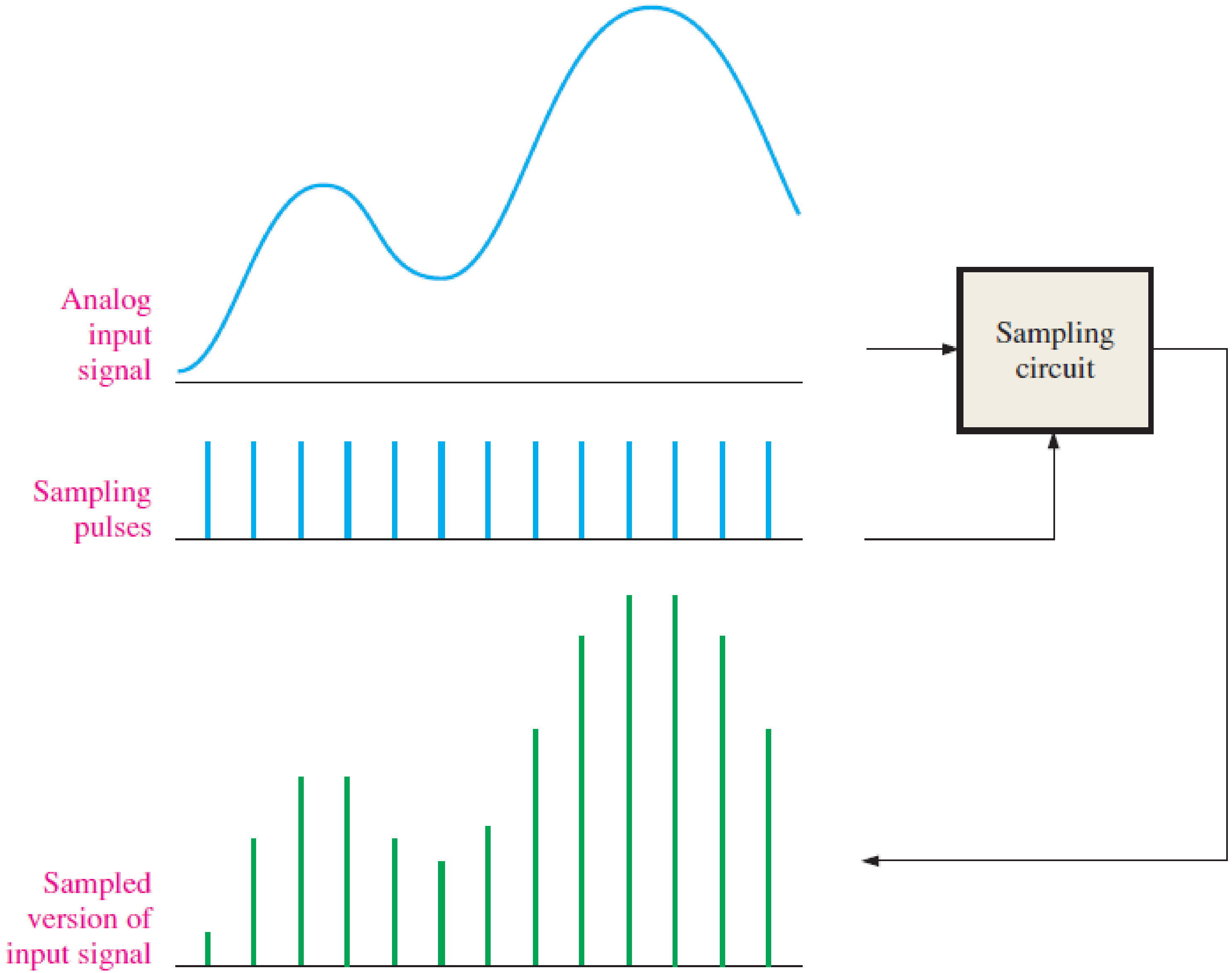
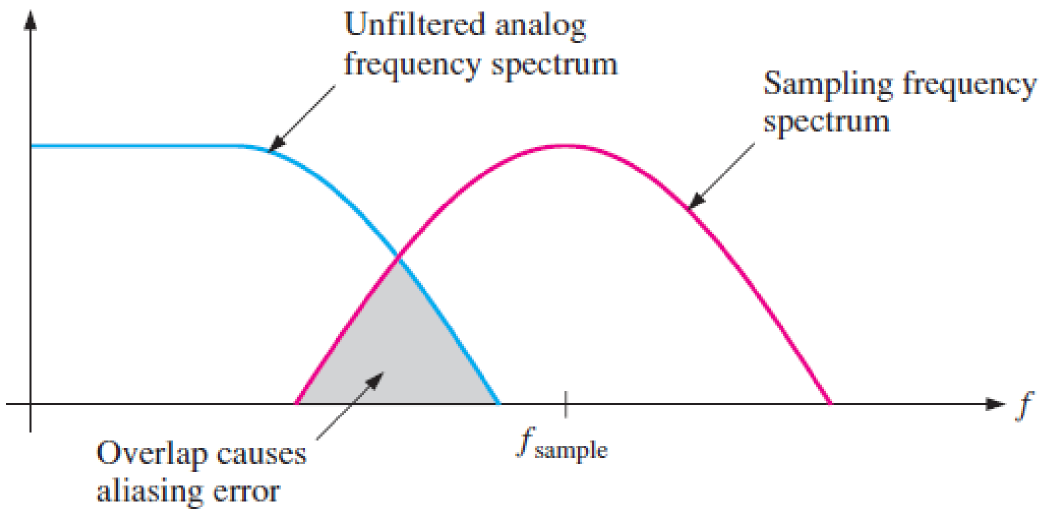



**Illustration of sampling process**

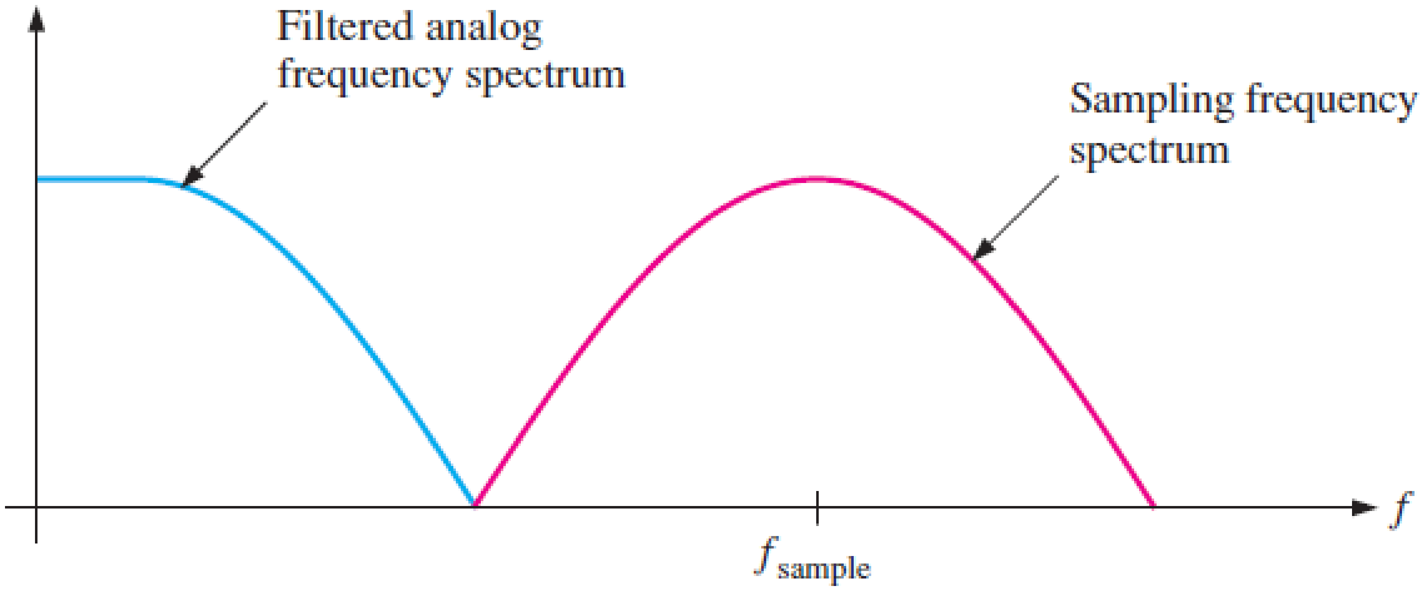
# Sampling and Filtering

- The reason we need a low pass filter, is to remove all frequency components (harmonics) of the analog signal that exceed the Nyquist frequency.
- If there are any frequency component in the analog signal that exceed the Nyquist frequency, an unwanted condition know as aliasing will occur.
- An alias is a signal produced when the sampling frequency is not at least twice the signal frequency.
- An alias signal has a frequency that is less than the highest frequency in the analog signal being sampled and therefore falls within the spectrum or frequency band of the input analog signal causing distortion. Such a signal is actually "posing" as the part of the analog signal when its isn't, thus the term alias.



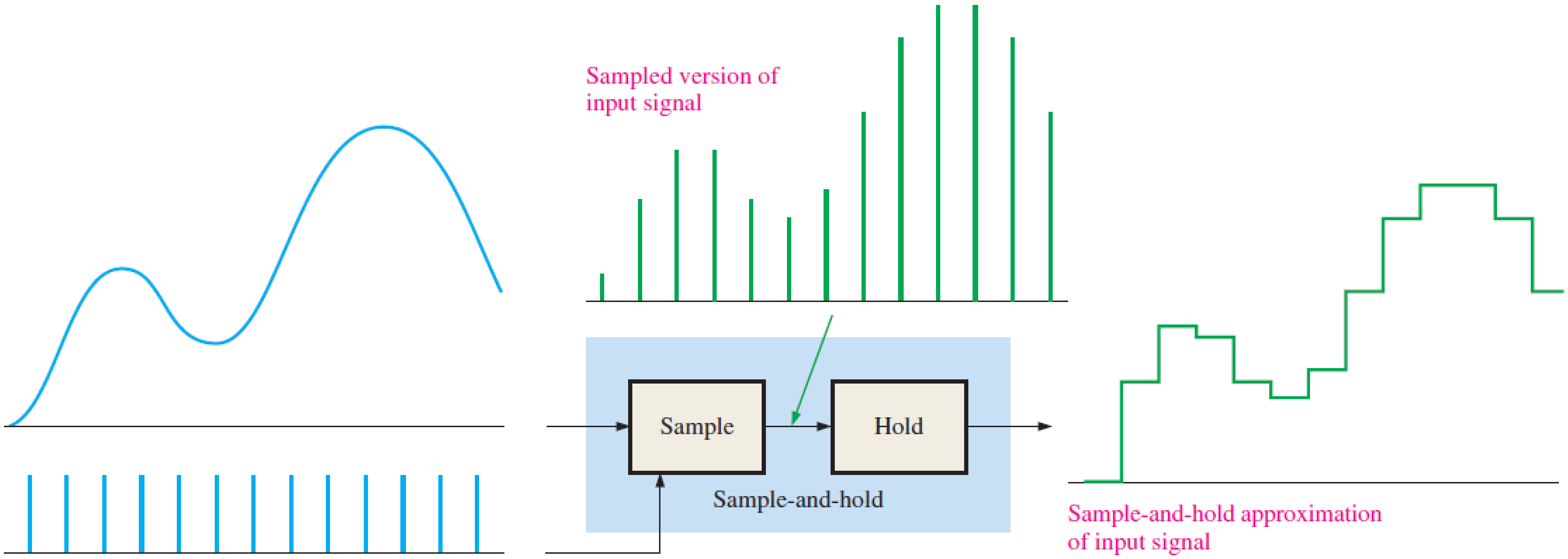
# Sampling and Filtering

- A low-pass anti-aliasing filter must be used to limit the frequency spectrum of the analog signal for given sample frequency.
- To avoid aliasing error, the filter must at least eliminate all analog frequencies above the minimum frequency in the sampling spectrum.



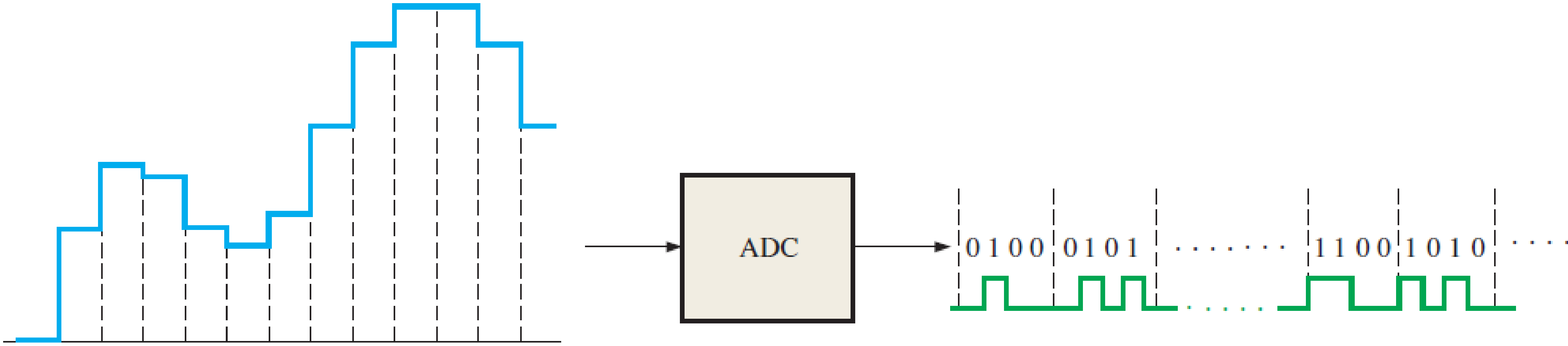
# Hold Operation

- The holding operation is part of the sample and hold block shown in figure. After filtering and sampling, the sampled level must be held constant until the next sample occurs.
- This is necessary for the ADC to have time to process the sampled value. This sample and hold operation results in a “stairstep” wave form that approximate the analog input waveform.



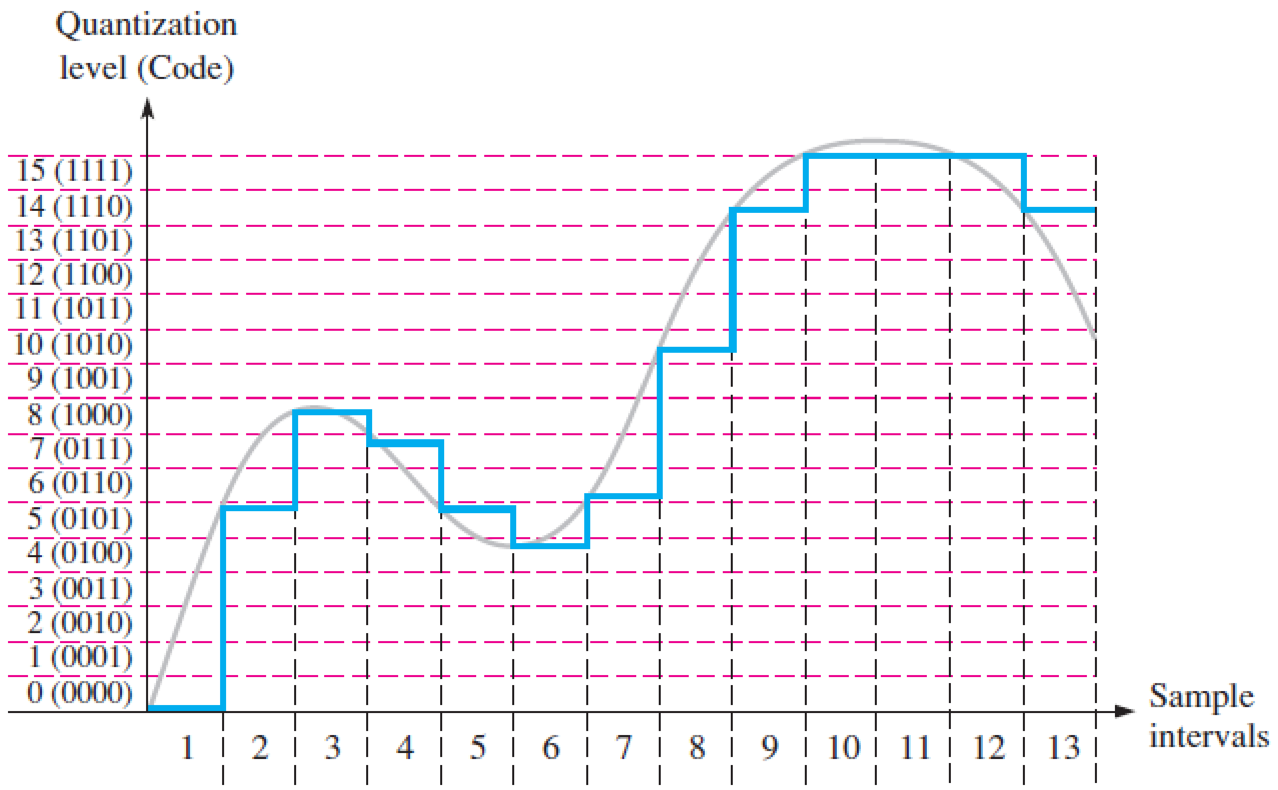
# Analog to Digital Conversion

- Analog to digital conversion is the process of converting the output of the sample and hold circuit to a series of binary codes that represent the amplitude of the input at each of the sample times.
- The sample and hold process keeps the amplitude of the analog input signal constant between sample pulses.
- Therefore, the analog to digital conversion can be done using a constant value rather than having the analog signal change during conversion interval, which is the time between the sample pulses.



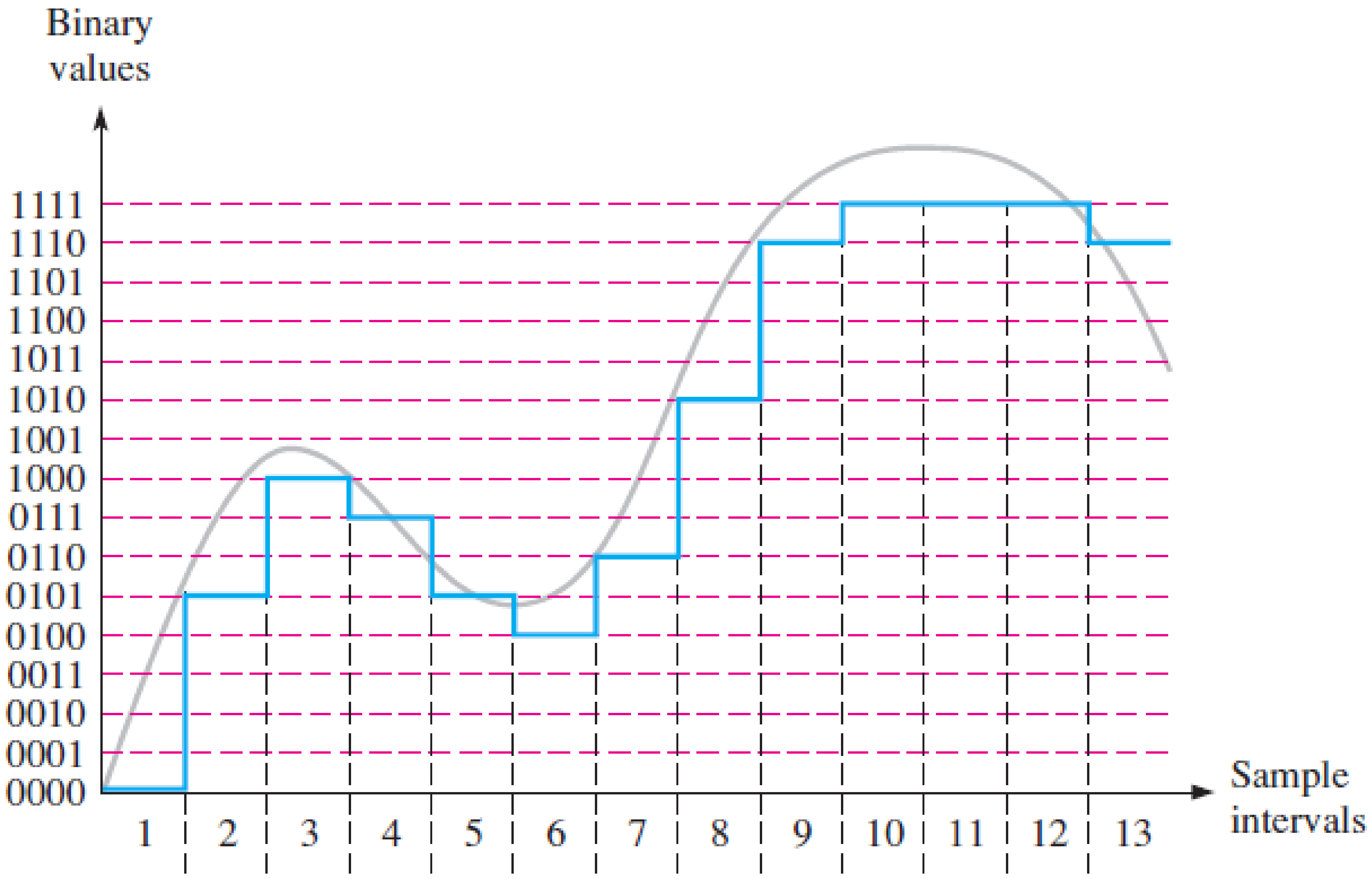
# Quantization

- The process of converting an analog value to a code is called quantization.
- During quantization process, the ADC converts each sampled value of analog signal to a binary code.
- The more bits that are used to represent a sampled value, the more accurate is the representation.




| Sample Interval | Quantization Level | Code |
|---|---|---|
| 1 | 0 | 0000 |
| 2 | 5 | 0101 |
| 3 | 8 | 1000 |
| 4 | 7 | 0111 |
| 5 | 5 | 0101 |
| 6 | 4 | 0100 |
| 7 | 6 | 0110 |
| 8 | 10 | 1010 |
| 9 | 14 | 1110 |
| 10 | 15 | 1111 |
| 11 | 15 | 1111 |
| 12 | 15 | 1111 |
| 13 | 14 | 1110 |

# Quantization



# Quantization

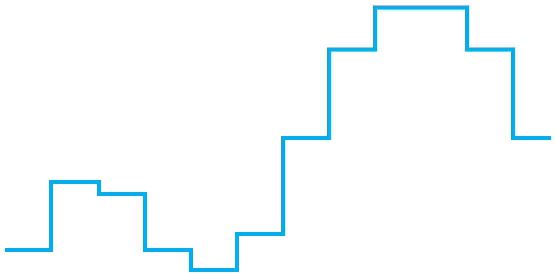
**For n-bit quantization:**

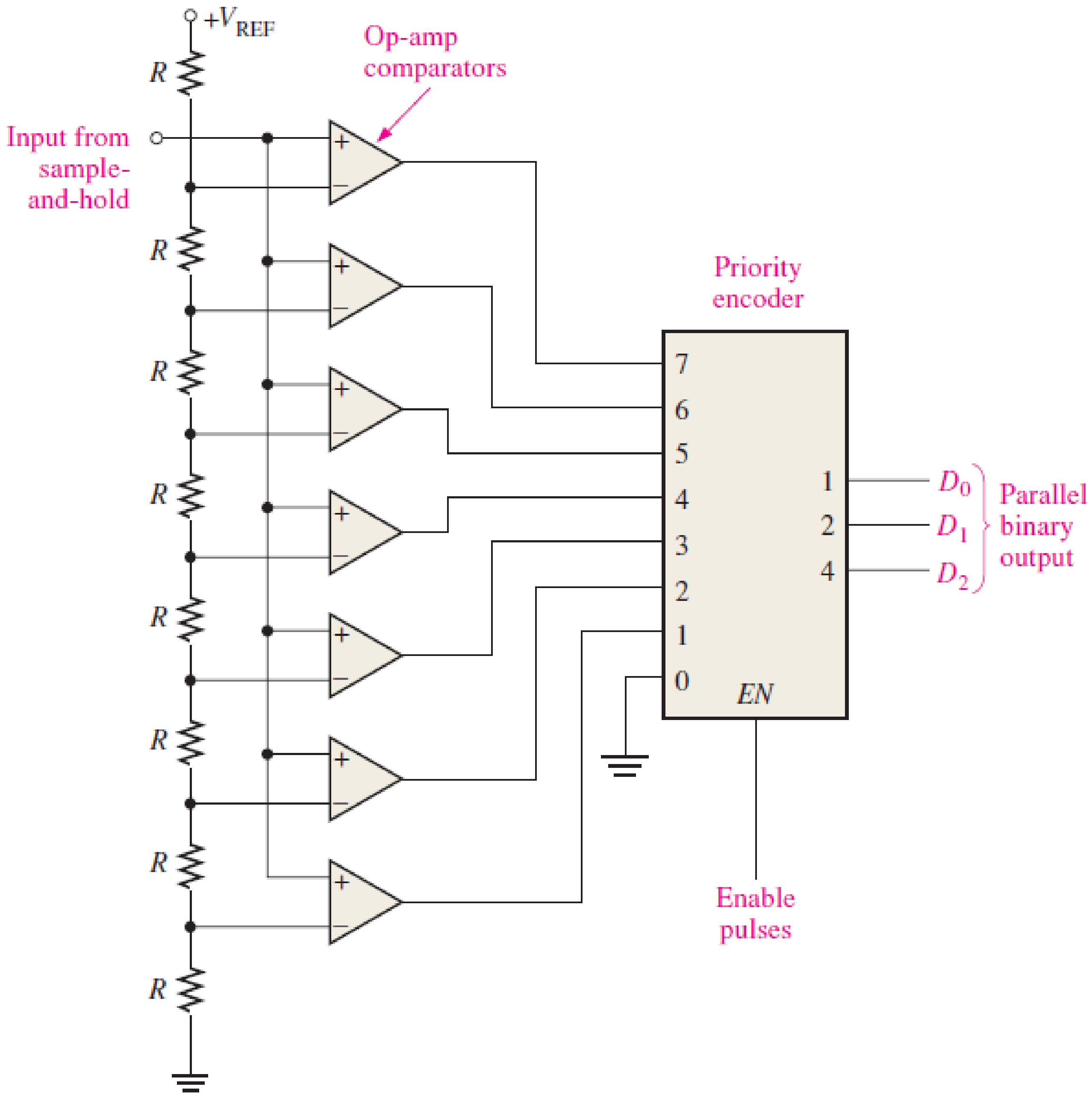
- No. of quantization steps: $\mathbf{2^n}$
- Full-Scale Input Voltage Range, FSR: $\mathbf{V_{i(max)} - V_{i(min)}}$
- Resolution: $\mathbf{\frac{FSR}{2^n}}$
- Digital Code, D: $\mathbf{\frac{Input\ Voltage - V_{i(min)}}{Resolution}}$
- **Example Problem:** An 8-bit ADC is capable of accepting an input voltage of range 0 to 10V. Find:
  a) The Resolution
  b) Digital output code for an input 5.2V.

# Flash Analog to Digital Converter

- The flash method utilizes comparators that compare reference voltages with analog input voltages.
- When the input voltage exceeds the reference voltage for a given comparator, a HIGH is generated.
- A 3-bit converter that uses seven comparator circuits; a comparator is not needed for all 0's condition.
- In general $2^n - 1$ comparators are required for conversion to an n-bit binary code.
- The number of bits used in an ADC is its resolution.
- The large number of comparators necessary for a reasonable-sized binary number is one of the disadvantage of the flash ADC.
- **Its chief advantage is that it provides a fast conversion time because of a high throughput, measured in samples per second (sps).**

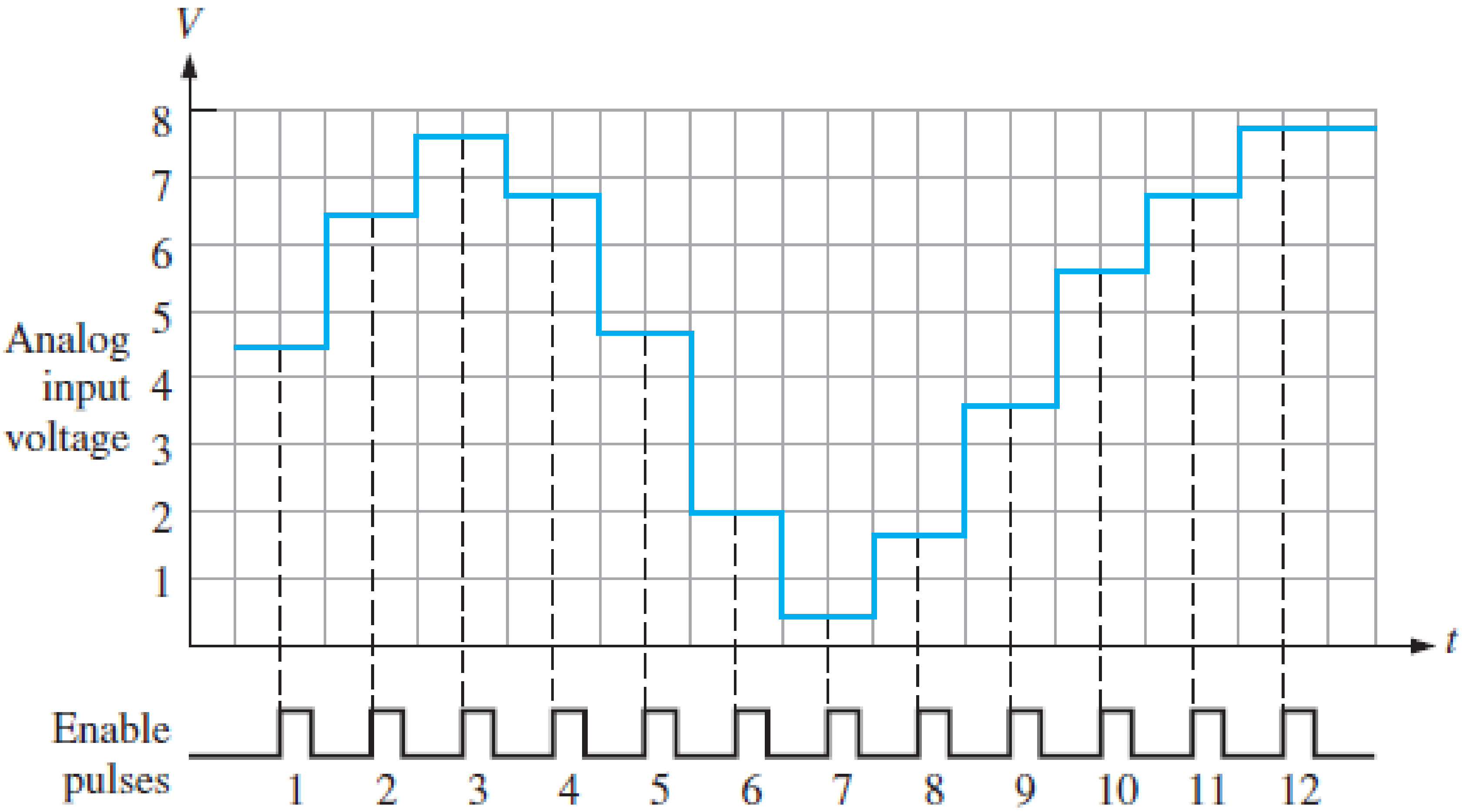
# Flash Analog to Digital Converter





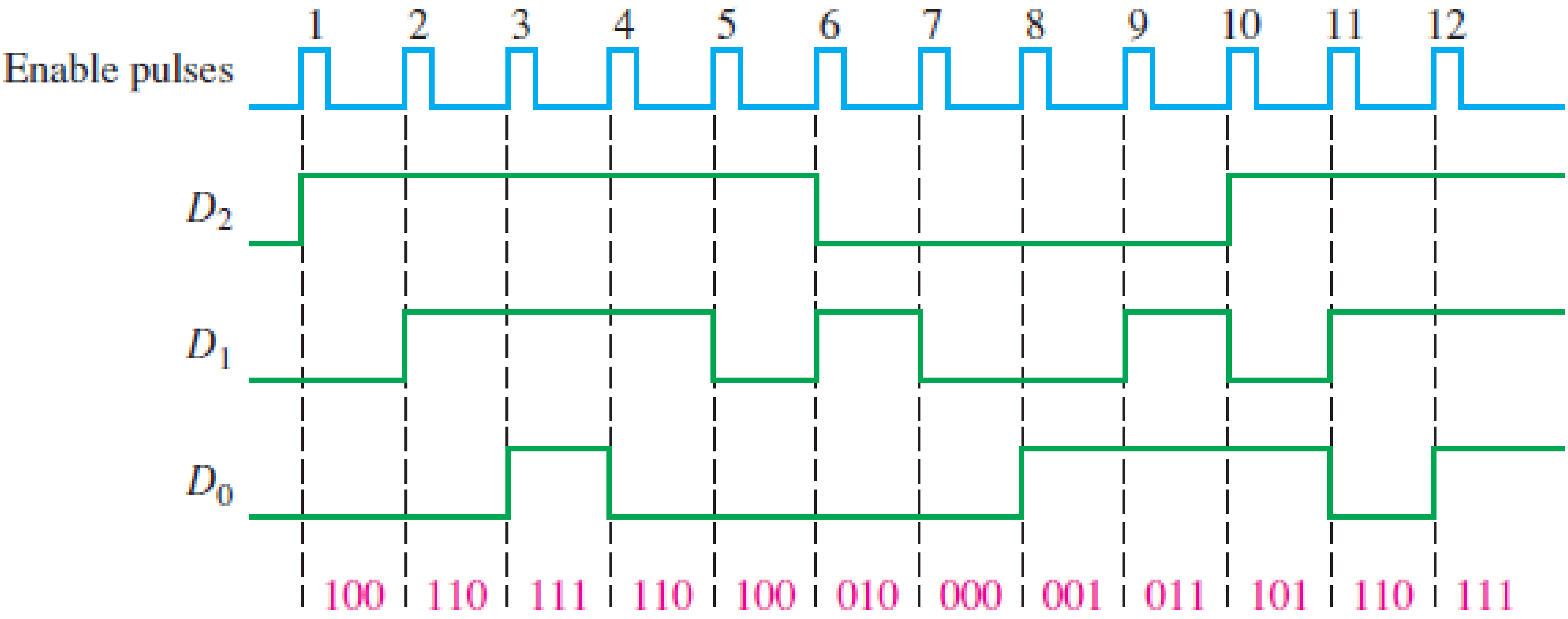
# Flash Analog to Digital Converter

Determine the binary code output of a 3-bit ADC for the input signal and encoder enable pulses shown. For this example $V_{REF} = +8V$



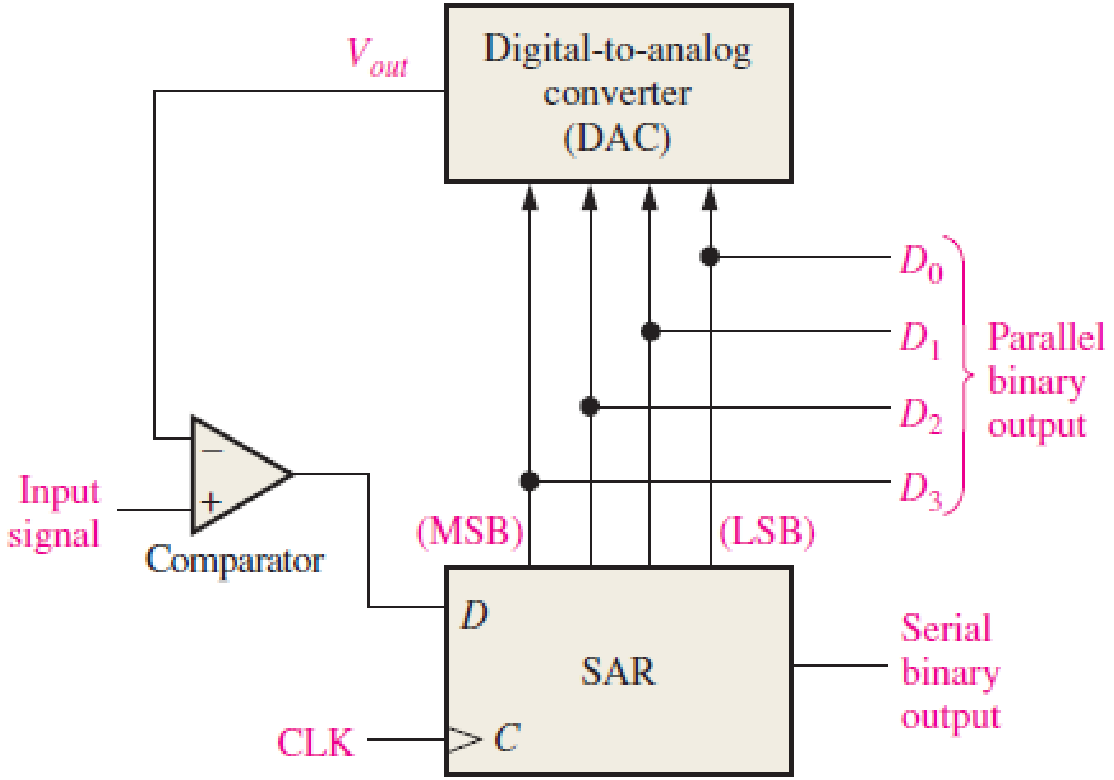
# Flash Analog to Digital Converter

**SOLUTION**



# Successive-Approximation Analog to Digital Converter

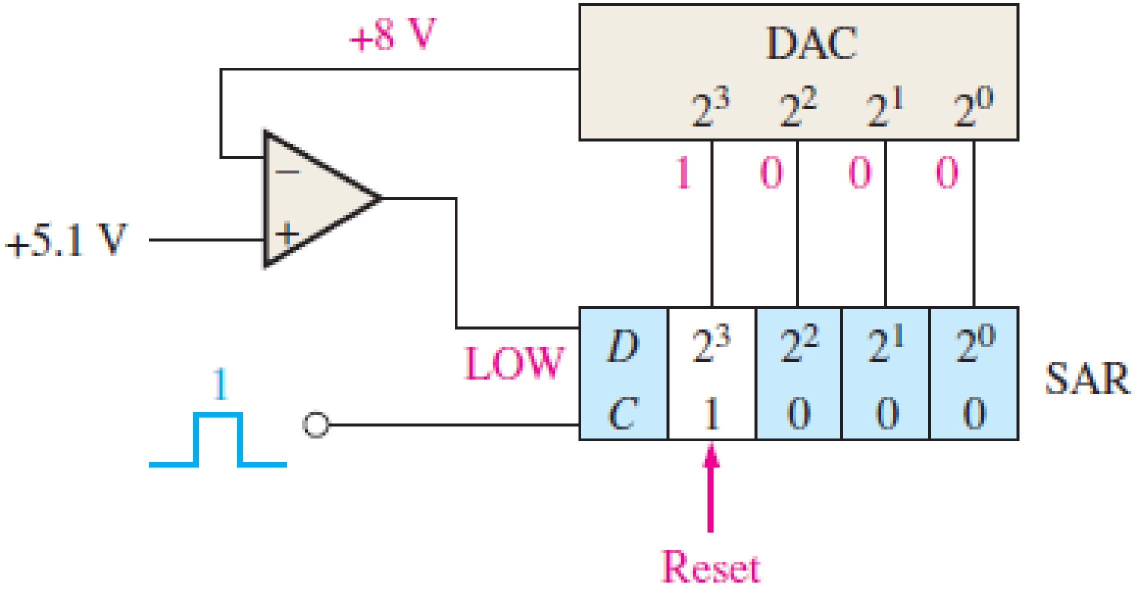
- Successive-Approximation is the most widely used ADC method.
- It is faster than a dual-slope converter.
- However, it is slower than a Flash ADC.
- It has a fixed conversion time for any value of analog input.
- An n-bit converter takes n cycles or steps to converts any analog input.



# Successive-Approximation Analog to Digital Converter

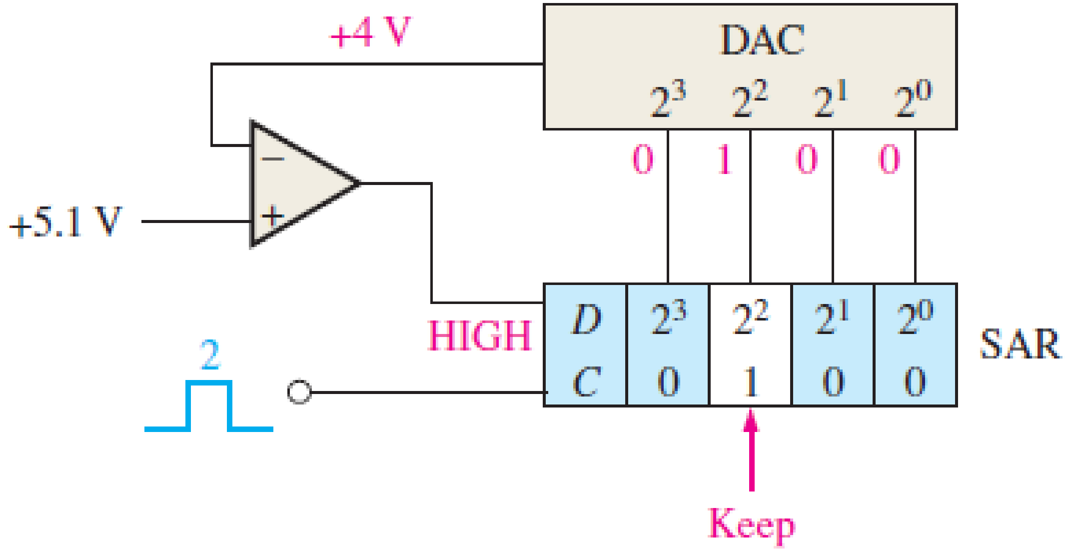
- The DAC outputs 8V for $2^3$ bit, 4V for $2^2$ bit, 2V for $2^1$ bit and 1V for $2^0$ bit.
- Now as an example we try to convert an input voltage of 5.1V

At the first pulse, the MSB is turned HIGH. As 8V is greater than 5.1V, it is therefore reset.

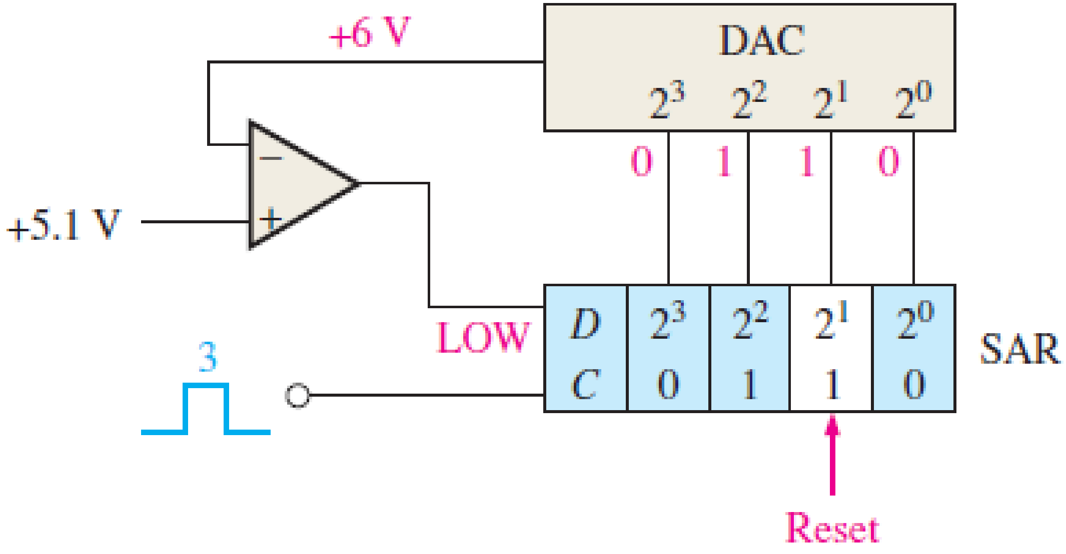


# Successive-Approximation Analog to Digital Converter

Then the next bit is turned HIGH. The corresponding output is 4V. As it is less than 5.1V, the bit is kept as HIGH.

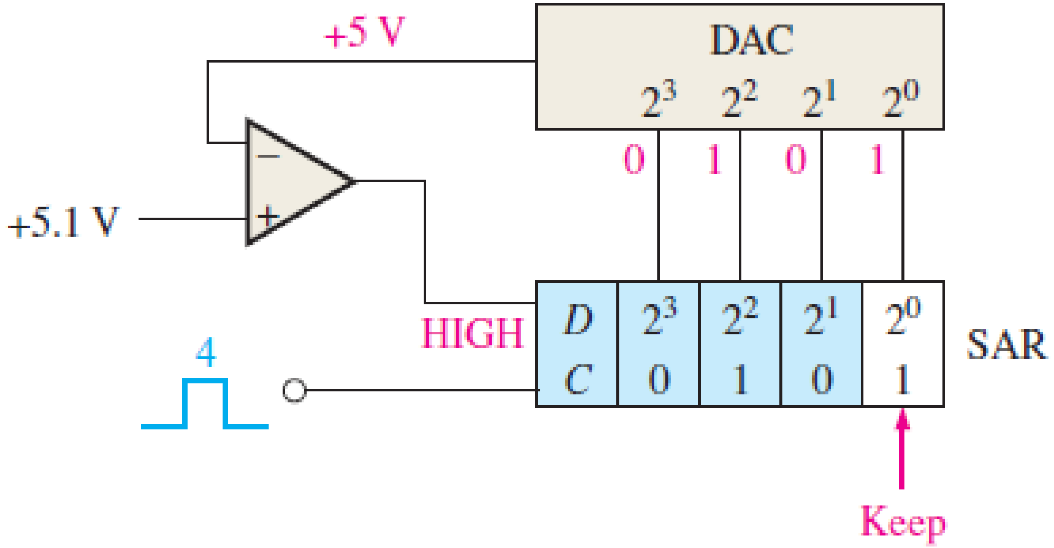



Then the next bit is turned HIGH, the corresponding output now is 6V. As 6V is greater than 5.1V, the bit is RESET.



# Successive-Approximation Analog to Digital Converter

Now, finally the last bit is turned HIGH. The corresponding output is 5V. As this is less than the input 5.1V, the last bit is retained as 1.




So after 4 cycles all the bit has been tried. Therefore, the digital code corresponding to 5.1V is 0101, which is 5V. If the no. of bit is increased the resolution of ADC will increase and thus will produce a more accurate approximation.

# References


1. Thomas L. Floyd, “Digital Fundamentals” 11th edition, Prentice Hall – Pearson Education.


# Thank You