Introduction to DSP

- Digital Signal Processing (DSP) is an application of analog and digital electronic technologies that deals with the processing of analog and digital signals by digital means.
- Generally, DSP includes the conversion of analog signals to digital equivalents, and the processing of these digital equivalents by digital operations such as storage, addition and multiplication by constants.
- The “results” of digital signal processing are, in some applications, converted to analog form.

Fundamental DSP Concepts

- Continuous-time Signals
- Discrete-time Signals
- Digital Signals
- Sampling
- Quantization
- Digitizing

Examples of Continuous-time Signals

- signals defined over a continuous range of time values
- continuous-time signals are generated by sensors and by test equipment
- continuous-time signals that are defined over a continuous range of amplitude values are referred to as analog signals

Continuous-time Signals

- signals defined only for particular values of time; may be continuous in amplitude
- discrete-time signals are generally obtained from continuous-time signals through the process of sampling
- discrete-time signals are represented by indexed (i.e. 0, 1, 2,...) sequences of numbers

Discrete-time Signals
Examples of Discrete-time Signals

Discrete-time signals are defined only for particular values of time; may be continuous in amplitude. Discrete-time signals are generally obtained from continuous-time signals through the process of sampling. Discrete-time signals are represented by indexed (i.e., 0, 1, 2, ...) sequences of numbers.

Continuous-time signals are defined over a continuous range of time values. Continuous-time signals are generated by sensors and by test equipment. Continuous-time signals that are defined over a continuous range of amplitude values are referred to as analog signals.

Signals

Digital Signals

- Digital signals are discrete-time signals, that are represented by a finite-length binary code whose amplitude is limited to a finite number of values.
- Digital signals are the typical signals used in data communications and in digital signal processing.

Sampling

- The process of selecting part of a continuous-time signal for conversion to a binary code.
- Sampling is performed during very short time intervals and is generally repeated at a constant rate.
- Sampling produces a sequence of amplitude values, i.e., a sequence of numbers.

An example of sampling

Q. What is the sampling frequency (rate) in this case?
A. Consider the number of samples in the discrete time signal derived from the 1 Hz analog signal: 30 samples indicates that the sampling rate is 30 Hz, or, more appropriately, 30 samples/second.

Sampling at a given rate

\[ f_{\text{sampling}} = 25 \times f_{\text{signal}} \]
Sampling at a lower rate

\[ f_{\text{sampling}} = 5 \times f_{\text{signal}} \]

Quantization

The process of representing an acquired analog sample by the nearest amplitude level that exactly corresponds to an integer scale of values.

Quantization means that a continuous range of amplitudes is converted to a finite number of distinct amplitudes (levels). If the actual sample value is 2.7 volts and the nearest levels on an integer scale are 2 volts and 3 volts, the quantized sample value is set to 2 volts.

The number of discrete levels is \(2^n\) where \(n\) is the number of bits. For example, quantization to 4 bits leads to \(2^4 = 16\) levels, quantization to 8 bits leads to \(2^8 = 256\) levels.

There are no special electronic circuits that perform quantization. Quantization takes place in the analog-to-digital converter (ADC).

Quantization Example

Quantization to 3 bits (i.e., 8 levels) is used to "cover" an amplitude range of 0 to 8 volts. Therefore, each quantization "step" is one volt. The amplitude values, resulting from sampling an analog signal in this amplitude range, as well as the associated quantized values, are given in the following table:

<table>
<thead>
<tr>
<th>sample amplitude</th>
<th>quantized value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.04</td>
<td>0</td>
</tr>
<tr>
<td>7.38</td>
<td>7</td>
</tr>
<tr>
<td>4.78</td>
<td>4</td>
</tr>
<tr>
<td>2.95</td>
<td>2</td>
</tr>
<tr>
<td>0.97</td>
<td>0</td>
</tr>
<tr>
<td>0.01</td>
<td>6</td>
</tr>
</tbody>
</table>

Quantization with full-scale input

\[ 2^4 = 16 \text{ levels} \]

(All 16 levels realized here)

Quantization with less than full-scale input

\[ 2^2 = 16 \text{ levels} \]

(8 levels realized here)
Quantization with more bits, i.e. more levels

$2^n = 256$ levels (maximum not realized here)

Continuous signal

Discrete values

Quantization error, with fewer bits

Continuous signal

Discrete values

With more bits, lower quantization error

Continuous signal

Discrete values

Quantization Error, Noise Voltage, and Signal to Noise Ratio (SNR)

$$V_{\text{noise(rms)}} = \frac{V_{\text{full-scale}} \times 0.289}{2^n}$$

$n = \text{the number of bits}$

Maximum signal to noise ratio (dB) = $6.02n + 1.76$

Digitizing

• generating the binary code "equivalent" of the amplitude of the continuous-time sample
• the process of digitizing is performed by the analog-to-digital converter (ADC)
• analog-to-digital converters are rated in terms of the number of output bits, the conversion speed and the accuracy of converting analog values to binary values

Analog-to-Digital Converters (ADC)

Important ADC Parameters

- **linearity**: the extent to which an output voltage increment that corresponds to a given binary input increment, is constant over the entire range of binary input values; linearity can be determined by measuring the output voltage changes corresponding to a binary input changes of one LSB, at various points in the binary input range, e.g. for a 3-bit system, at 000, 100 and 111.

- **dynamic range**: the ratio of the largest input value for which linearity is maintained, to the smallest input value that produces a measurable output.
Digital-to-Analog Converters (DAC)

**Important DAC Parameters**

**linearity**
the extent to which an output voltage increment that corresponds to a given binary input increment, is constant over the entire range of binary input values; linearity can be determined by measuring the output voltage changes corresponding to a binary input changes of one LSB, at various points in the binary input range, e.g. for a 3-bit system, at 000, 100 and 111.

**dynamic range**
the ratio of the largest input value for which linearity is maintained, to the smallest input value that produces a measurable output.

DAC Implementation

**Binary-Weighted Resistors DAC**

The circuit is conceptually simple, but it is difficult to implement because of the required 1, 2, 4, 8, … resistor values ratio.

**R-2R DAC**

The circuit, based on a R-2R ladder network, requires only two resistor values: R and 2R, and can be implemented in two ways:
- a) Switched Voltage R-2R DAC
- b) Switched Current R-2R DAC

Multiplying vs. non-multiplying D/A Converter

“non-multiplying”
DAC output = binary input x internal fixed reference voltage or current

“multiplying”
DAC output = binary input x external adjustable reference voltage or current

Transducers and Sensors

**Transducers** – devices that convert from one physical quantity to another, e.g. temperature to electrical, pressure to electrical, light to electrical

**Sensors** – transducers that produce electrical signals for measurement and control applications

the terms transducer and sensor are sometimes used interchangeably to describe any source that produces signals resulting from the conversion of physical quantities to an electrical equivalent

Converting a real-world signal (sound) to an electrical signal
Basic (Conceptual) DSP System

additional circuits are usually required to implement practical DSP systems

Practical DSP System

Practical DSP systems also include the following:

• an analog low-pass filter to "bandlimit" the analog input in order to prevent aliasing
• a sample-and-hold (S/H) circuit to "hold" the sample until the ADC completes the conversion of the sample
• an analog low-pass filter to "reconstruct" the output of the DAC into a "clean" analog signal, by removing the unwanted high frequency signals created by the DAC operation

Signal Analysis DSP System

examples: touch-tone dialing; modulation analysis; speech recognition; digital oscilloscope; PCM encoder

Signal Synthesis DSP System

examples: digital synthesis of complex analog signals, such as music and radar test signals; DVD player; text-to-speech conversion; PCM decoder

Digital-only DSP System

examples: non-real-time analysis of stored data such as stock market prices, temperature records and geophysical data
## Analog Processing Systems

- Inflexible: system configuration and parameters can be modified only by re-arranging and/or by changing components or component values
- Sensitive to variations caused by temperature, component tolerances, and age effects
- Simple and relatively inexpensive to develop; e.g., active filters based on operational amplifiers
- Limited mathematical background is sufficient for the development of analog signal processing systems

## Digital Processing Systems

- Flexible: system configuration and parameters can be modified through software modifications; systems that have no analog equivalent can be implemented.
- Insensitive to variations caused by temperature, component tolerances, and age effects
- Complex and relatively expensive to develop; special software and hardware are usually required
- Strong mathematical background is required in order to fully realize the benefits of using digital signal processing techniques

## Analog vs. Digital Signal Processing

### Analog
- Inflexible: system configuration and parameters can be modified only by re-arranging and/or by changing components or component values
- Sensitive to variations caused by temperature, component tolerances, and age effects
- Simple and relatively inexpensive to develop; e.g., active filters based on operational amplifiers
- Limited mathematical background is sufficient for the development of analog signal processing systems

### Digital
- Flexible: system configuration and parameters can be modified through software modifications; systems that have no analog equivalent can be implemented.
- Insensitive to variations caused by temperature, component tolerances, and age effects
- Complex and relatively expensive to develop; special software and hardware are usually required
- Strong mathematical background is required in order to fully realize the benefits of using digital signal processing techniques

## DSP Applications

- Test Instrumentation
- Industrial Robotics
- Medical Electronics
- Voice Recognition
- Telephone Systems
- Image Processing
- Computer Graphics
- Remote Sensing
- Geophysical Data Analysis
- Radar Signal Processing
- Sonar Signal Processing
- Meteorology
- Control Systems
- Speech Synthesis
- Digital Communications
- Data Communication
- Digital Audio Systems
- Sound/Music Effects
- Digital TV Systems
- Data Compression
- Navigation Systems
- Financial Data Analysis
- Data Security

## Classification of Signals

- Continuous-time vs. discrete-time
- Periodic vs. non-periodic
- Infinite vs. finite length (or, duration)
- Deterministic vs. Stochastic

## The Sampling Process

The conversion of a CT signal to a digital signal, which can be processed by DSP techniques, starts with the sampling of the CT signal.

**Sampling**: The process of “capturing” the amplitude of a continuous-time signal at selected instants in time. The captured samples form a discrete-time signal with a continuous range of amplitude values.

The process of sampling implies the measurement of the amplitude of the continuous-time signal at specific instants in time.

Two cases must be considered: slow-varying continuous-time signals and fast-varying continuous-time signals. The terms “slow-varying” and “fast-varying” relate to the amplitude variation of the signal during the presence of the sampling pulse.

Small amplitude variations are associated with slow-varying signals and large amplitude variations are associated with fast-varying signals. Consider the following examples of analog signals to be sampled:

- **Slow-varying signals**
  - Gradual changes in amplitude.
  - Examples: voltages or currents in a circuit over a short period.

- **Fast-varying signals**
  - Rapid and frequent changes in amplitude.
  - Examples: electrical signals in a high-speed communication system.
**Sampling Slow-varying and Fast-varying Signals**

- Slow-varying signal
- Fast-varying signal

Note the importance of narrow sampling pulses and of a fast sampling rate, particularly for fast-varying signals.

**The Sample-and-Hold (S/H) Requirement**

For "slow-varying" continuous-time signals, the A/D converter can perform the sampling without the use of an external sampling circuit. The typical A/D converter is sufficiently fast to complete the digitizing process before the amplitude of the input signal changes.

For "fast-varying" continuous-time signals, it is usually necessary to sample the analog signal and hold the "captured" amplitude value so that the A/D converter, which is usually the slow link in the chain, can complete the digitizing process while the amplitude of the input signal changes.

A special circuit referred to as "Sample and Hold" (S/H) performs this combined operation. The S/H circuit "freezes" the input analog signal until the A/D converter completes the digitizing of the sample.

**The Sample-and-Hold (S/H) Requirement**

The S/H is used to "hold" the value of the analog sample until the ADC completes the conversion of the sample to a binary number.

**Aliasing**

Aliasing is said to occur when sampling one, two or more signals, of different frequencies, produces samples of the same amplitudes; two types of aliasing can be defined.

**Aliasing Type I**

- Signal A is the "desired" signal and signal B is the "unwanted" resulting signal.
- The sampling rate is too low, compared to the signal's maximum frequency.
- As a result, the samples appear to represent a lower frequency signal that does not really exist.
- To prevent this type of aliasing, the sampling rate should be increased, to be at least twice the highest input signal frequency.

**Aliasing Type II**

- Signal B is the "desired" signal and signal A is the "unwanted, external" higher frequency signal.
- "Unwanted, external" higher frequencies are present together with the "wanted" signal.
- As a result, the high frequency signal masquerades as the low frequency signal.
- To prevent this type of aliasing, a low-pass filter is used, at the input of the DSP system, to remove or attenuate the "unwanted" higher frequency signal; this is known as "bandlimiting" the input signal.
**Aliasing**

**Aliasing Type I**
- Signal A is the "desired" signal and Signal B is the "unwanted" signal.
- The sampling rate is too low, compared to the signal's maximum frequency.
- As a result, the samples appear to represent a lower frequency signal that does not really exist.
- To prevent this type of aliasing, the sampling rate should be increased, to be at least twice the highest input signal frequency.

**Aliasing Type II**
- Signal B is the "desired" signal and Signal A is the "unwanted" higher frequency "aliased" signal.
- "Unwanted" higher frequencies are present together with the "wanted" signal.
- As a result, the high frequency signal masquerades as the low frequency signal.
- To prevent this type of aliasing, a low-pass filter is used, at the input of the DSP system, to remove or attenuate the "unwanted" higher frequency signal; this is known as "bandlimiting" the input signal.

**The Nyquist frequency**
- \( \frac{1}{2} \) of the sampling frequency \( f_S \)
- Also called the Nyquist limit.
- Signal frequencies > \( f_S / 2 \), if sampled, will not be properly represented \( \implies \) aliasing will occur.

**Signal freq. to be sampled is below the Nyquist limit**
- No aliasing occurs.

**Signal freq. to be sampled is above the Nyquist limit**
- The 3200 Hz signal has been aliased to 200 Hz.

**Example of Aliasing**
- The following signal frequencies produce identical samples when sampled at \( f_S = 3000 \) Hz:
  - 200 Hz
  - 3200 Hz
  - 6200 Hz
  - 9200 Hz
  - 12200 Hz

  An unwanted signal of frequency \( f = f_{signal} + N \times f_s \) will produce identical samples, where \( N \) is a positive integer (0, 1, 2, 3, 4, 5 ...).

  Therefore, the optimum condition is that \( f_s \geq 2 \times f_{signal} \) i.e. sampling at twice the Nyquist rate, or higher, to prevent aliasing.

**Example-related Operations**

**Example no. 1**
- Consider an "input" sequence of numbers: 1, 4, 7, 3, 9, 2, 4, 1, 6, 2.
- Add the first two numbers and divide the result by two:
  \[ 1 + 4 = 5 \text{ and } 5 / 2 = 2.5 \]
- Repeat this process for the second and third numbers:
  \[ 4 + 7 = 11 \text{ and } 11 / 2 = 5.5 \]
- Repeat for all relevant pairs:
  \[ 7 + 3, 3 + 9, 9 + 2, \text{ etc.} \]
- The result is an "output" sequence of numbers: 2.5, 5.5, 5, 6, 5.5, 3, 2.5, 3.5, 4.
**Example no. 2**
Consider the same “input” sequence of numbers: 1, 4, 7, 3, 9, 2, 4, 1, 6, 2.
Subtract the first two numbers and divide the result by two:
\[1 - 4 = -3 \text{ and } -3/2 = -1.5\]
Repeat this process for the second and third numbers:
\[4 - 7 = -3 \text{ and } -3/2 = -1.5\]
Repeat for all relevant pairs:
7 and 3, 3 and 9, 9 and 2, etc.
The result is a second “output” sequence of numbers:
-1.5, -1.5, 2, -3, 3.5, -1, 1.5, -2.5, 2

**DSP-related Operations**

Can you identify an interesting property of the “output” sequence in Example no.2?
What mathematical procedure is illustrated in these two examples?
Note that the subtraction of two numbers is actually an addition where one of the numbers is negative: \[7 - 4 = 7 + (-4) = 3\]
Note a similar relation between multiplication and division: \[\frac{4}{3} = 4 \times \frac{1}{3}\]
Recall how computers perform arithmetic operations such as addition and multiplication.
Some computers have a hardware multiplier while others, such as the 68HC11, can perform multiplication only through special software.

**Computer Multiplication and Division**

Given a binary number stored in a computer register:
\[
\begin{array}{c|c|c|c|c|c|c|c|c|c|c}
  & MSB & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\
  & LSB & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]
Shifting the number left by one bit → Multiplication by 2
Shifting the number right by one bit → Division by 2
Shifting left by 2,3,4,…. bits → Multiplication by 4,8,16,…
Shifting right by 2,3,4,…. bits → Division by 4,8,16,…

**Fourier Synthesis**

Signals can be “viewed” in the time domain, and in the frequency domain via analysis based on Fourier theory

**Fundamental DSP Tasks**

- **Signal Analysis**
  - Determining the properties of signals
  - Spectrum (frequency/phase) analysis
  - Detection & recognition, e.g. radar, sonar, speech recognition, face recognition

- **Signal Filtering**
  - Modifying signals based on parameters such as frequency or spatial location
  - Removal of noise or interference
  - Separation of frequency bands

- **Signal Analysis followed by Filtering**
- **Signal Filtering followed by Analysis**