WHAT IS A SIGNAL?

In general terms, a signal is an action, movement, or sound that gives information, a message, a warning, or an order. A signal could also be referred to as a physical quantity that varies with time, space or any other independent variable by which information can be conveyed.
In electronics, a signal is an electric current or electromagnetic field used to convey data from one place to another. The simplest form of signal is a direct current (DC) that is switched on and off.
In telephony, a signal is special data that is used to set up or control communication.
Some examples of modern high Speed signals are the voltage charger in a telephone wire, the electromagnetic field emanating from a transmitting antenna, variation of light intensity in an Optical fibre.

A signal as referred to in signal processing as a function that conveys information about the behaviour or attributes of some phenomenon. A signal may be a function of time, temperature, position, pressure, distance etc. Systematically we could define a signal as a function of one or more independent variables which contains some information. Many signals are naturally generated. However, few signals are generated synthetically.

For example, speech and music signals represent air pressure as a function of time at a point in space. A black and white picture is a representation of light intensity as a function of two spatial coordinates. The video signal in television consists of a sequence of images, called frames, and it's a function of three variables: two spatial coordinates and time.

A signal carries information, and the objective of signal processing is to extract useful information carried by the signal.

WHAT IS A FUNCTION?

A function is a relation between a set of inputs and a set of permissible outputs with the property that each input is related to exactly one output. The function is formed using an independent variable (i.e the input) and an expression which when evaluated gives the output for that particular independent variable.

A function is regarded as a signal when its independent variable(s) contain some information.

Examples of some functions include;

f(x) = x^2 where x is an independent variable that conveys the information describing the system

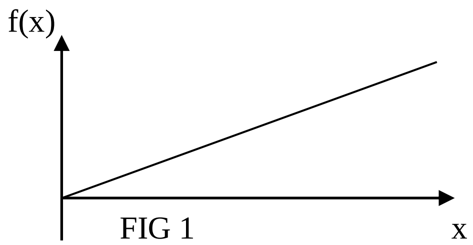


FIG 1

f(x) = sin w_0x

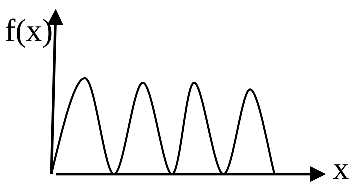


FIG 2

$f(x) = x^2 + xy^2 + y^3$

TYPES OF SIGNALS.

Based upon the nature and characteristics in the time domain, the signal may be broadly classified as under;

1. Continuous-time signals
2. Discrete-time signals

Continuous-time Signals

A continuous-time signal is defined as a mathematical continuous function. This function is defined continuously in the time domain. For continuous-time signals, the independent variable is time (t). A signal of continuous amplitude and time is known as a continuous signal or analog signal. A continuous time signal is represented by x(t) This signal will have some value at every instant of time. The electrical signals derived in proportion with physical quantities such as temperature, pressure, sound etc are generally continuous signals.

FIG 3: Continuous-time signal

Where for example the range of $t = 0 < t < \infty$

Discrete-time Signals

A discrete-time signal is defined only at certain time-instants. For discrete-time signals, the independent variable is time t. A discrete time signal is represented by x(t). If we take the blood pressure readings of a patient after every one hour and plot the graph, then the resultant signal will be a discrete time signal. A quantized discrete time signal is referred to as a digital signal.

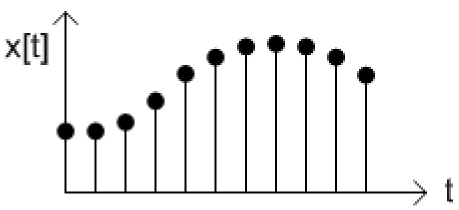


FIG 4: Discrete-time signal

Where for example $t = \pm 1, \pm 2, \pm 3$

The continuous-time and discrete-time signals can further be classified;

(i) Deterministic and non-deterministic (random) signals

(ii) Periodic and non-periodic signals

(iii) Even and Odd signals
(iv) Energy and power signals

DIMENSIONS OF SIGNALS

Dimensions define the minimum number of pointd required to point a position of any particular object within a space.

The dimension of a signal is determined by the number of independent variables its function depends on. Signals are broadly classified into two broad categories;

1. One dimensional signals
2. Multidimensional signals

One Dimensional Signals

When the function depends on a single variable, the signal is said to be one dimensional. A one dimensional signal is represented using two axes (amplitude vs. time). Example of one dimensional signal is speech signal whose amplitude varies with time. Also a waveform is a one dimensional signal.

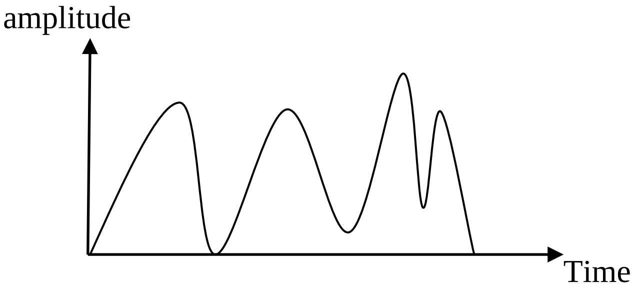


FIG 5: One dimensional signal

F(x) = waveform

Multidimensional Signals

When the function depends on two or more variables, the signal is said to be multidimensional. The example of a multidimensional signal is an image because it is a two dimensional signal with horizontal and vertical coordinates. The figure below depicts the image as a two dimensional signal with horizontal and vertical coordinates.

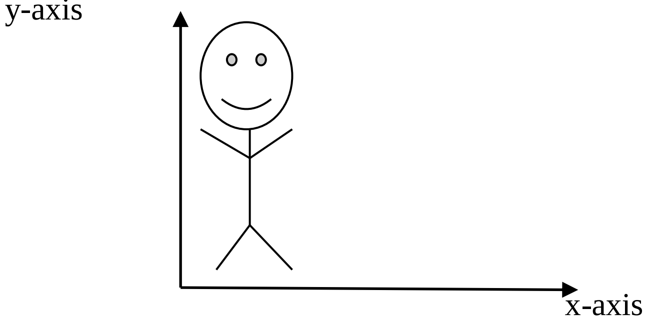


FIG 6: Two dimensional signal

WHY DO WE STUDY SIGNALS?

The importance of Signals is not restricted only to electrical engineering but it is as applicable to all other engineering disciplines, and indeed, in other seemingly unrelated fields such as seismology, economics, sociology, transportation, public and private administration, and political systems.

In computer engineering, signals are essential in the following;

1. Data communication
2. Behaviour of a system

Data communication

The successful transmission of data depends principally on two factors; the quality of the signal being transmitted and the characteristics of the transmission medium. In data transmission, the terms analog and digital correspond roughly to continuous and discrete respectively. Signals are electric or electromagnetic representations of data. In order to transmit data, signal is required to convey this data from place to another. Signal in this context acts as a data (information) carrier.

Behaviour of a system

Signals generally convey information about the state or behavior of a physical system. The study of signals, its classification and processing gives a great detail of system behavior. The behavior of a system is essential as it helps the designer know if the system is working as it should.

PRACTICAL EXAMPLES OF SIGNALS.

Remote controls for television

Remote controls for television sets do generate electrical signals. TV remotes are designed to set the channel accurately. They set the channel, adjust volume, etc. by sending signals to the TV. The remote control uses a line of sight (i.e the remote should physically point towards the television) as any obstruction in the line of sight causes a disconnect due to obstruction of the signals.

Connection for a phone

When you turn on your cell phone, the device begins searching for certain key frequencies the network uses for identification purposes. If it can connect to a tower on one of these frequencies, it sends a unique identifying number to the network, letting the system know your phone’s status and its current location. Once connected, your phone will continue to monitor this carrier frequency for incoming calls, and will use it to signal an outgoing call should you need to make one.

Sound

Since a sound is a vibration of a medium (such as air), a sound signal associates a pressure value to every value of time and three space coordinates. A sound signal is converted to an electrical signal by a microphone, generating a voltage signal as an analog of the sound signal, making the sound signal available for further signal processing.

Images

A picture or image consists of a brightness or color signal, a function of a two-dimensional location. The object's appearance is presented as an emitted or reflected electromagnetic wave, one form of electronic signal.

2.1 SIGNAL PROCESSING

A signal is anything that carries information, for example the human voice, gestures, chirping of birds, etc. these examples show forms of signals and they represent something (information).The act of finding out what these information mean brings us to the meaning of signal processing, in a layman term signal processing is the act of extracting the information carried by a signal.

Therefore signal processing can be said to be the mathematical representation of a signal and the algorithmic operation carried out on a signal to extract the information present.

WHY SHOULD WE PROCESS SIGNALS

Signals are processed because the following reasons:

i. To extract meaningful information from a signal
ii. To remove noise from the signal
iii. To make the signal transmittable
iv. To remove distortion of sample

For example an analog signal processing module converts real world information such as sound wave in the form of 0s and 1s to make it understandable and usable by the digital system such as biometric systems.

WAYS OF PROCESSING SIGNALS (ANALOG, DIGITAL, AND MIXED SIGNALS)

The basic ways through which signals can be processed are: Fourier transform, Laplace transform, Filters, and Z-transform.

FOURIER TRANSFORM

Fourier's Theorem essentially states that the frequency content of any signal can be described as the sum of a specific set of sine waves, this is because the sine wave is the only pure frequency and any distortion of this shape represents harmonics of some fundamental frequency. Thus any wave, no matter how oddly shaped, can be broken down into its component sine waves. Hence Fourier transform is simply a mathematical transformation that changes a signal from a time domain representation to a frequency domain representation thereby allowing one to observe and analyze its frequency content

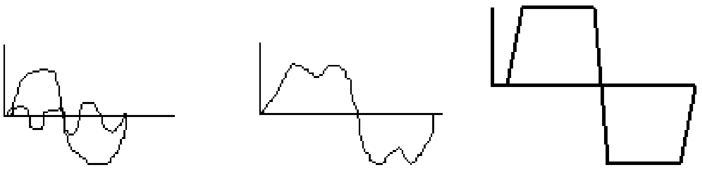


Fig 1a. b. c.

The diagram above shows a fundamental frequency with its 3rd harmonic. Combining them, we get a composite wave form that looks somehow like fig.1b. If we combine enough harmonics of increasing frequency and decreasing amplitude, we end up with a near perfect "square" wave(fig.1c).

LAPLACE TRANSFORM

The Laplace transform is another mathematical tool which is used for the analyses of signals and systems. In fact, the Laplace transform provides a broader characterization of the signals and systems compared to Fourier transform. The Laplace transforms uses damped waves through the use of an additional factor $e^{-\sigma}$ where σ is a positive number .furthermore, it is widely used for describing the continuous circuits and analyses at signal flow through causal linear-time invariant (LTI) systems with non-zero initial conditions.

FILTERS

Remember that the Fourier transform shows the frequency content of a signal. Filtering is the process of removing certain portions of the input signal in order to create a new signal. A familiar example would be the bass and treble controls on a CD player or electric guitar. There are four basic filter types:

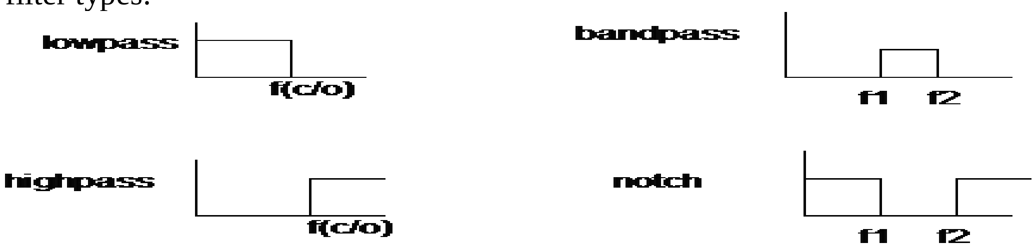


Fig 2

i. The Low pass filter which removes all frequencies above the cut off frequency. (Typically used for noise removal and data smoothing.)
ii. The High pass filter removes all frequencies below the c/o frequency. (Used for DC or low frequency drift.)
iii. Band pass filter removes all frequencies outside f 1 - f 2. (Used often in EEG measurements.)
iv. Notch filter removes all frequencies between f 1 and F 2. (60Hz noise removal).

A filter removes unwanted frequencies by sending the input signal through a system function H(s) which determines the degree of amplification for each frequency in the signal. The desired frequencies are boosted by the instrument gain while the unwanted frequencies are boosted by a gain of zero.

Z-TRANSFORM

The Z-transform is a useful tool and **is the most suited in the analysis of discrete-time signals** and systems and is the discrete-time counterpart of the Laplace transform for continuous-time signals and system. Z-transform has less limitation than the previous techniques. The z-transform may be used to solve constant coefficient difference equations, evaluate the response of a linear time-invariant system to a given input, design linear filters.

The z-transform of a discrete-time signal x(n) is defined by;

$X(z)=\sum_{n=-\infty}^{\infty} x(n)z^{-n}$

Where $z=re^{jw}$ is a complex variable.

DSP

DSP is an acronym that stands for Digital Signal Processing, which is the process of analyzing and modifying a signal to optimize or improve its efficiency or performance. It involves applying various mathematical and computational algotrithms to analog and digital signals to produce a signal that's of higher quality than the original signal.DSP is primarily used to detect errors, and to filter and compress analog signals in transit. It is a type of signal processing performed through a digital signal processor or a smilarly capable device that can execute DSP specific processing algorithms. Typically, DSP first converts an analog signal into a digital signal and then applies signal processing techniques and algorithms. For example, when performance on audio signals, DSP helps reduce noise and distortion. Some of the applications of DSP include audio signal processing, audio compression, video compression, digital communications, radar, sonar, seismology, digital image procesing, speech recognition, biomedicineand lots more. Specific examples are speech compression and transmission in digital mobile phones, room correction of sound in hi-fi and sound reinforcement applications, weather forecasting, economic forecasting, seismic data procssing, analysis and control of industrial processes, medical imaging such as CAT scans and MRI, MP3 compression, computer graphics, image manipulation, hi-fi loudspeaker crossovers and equalization, and audio effects for use with electric guitar amplifiers.

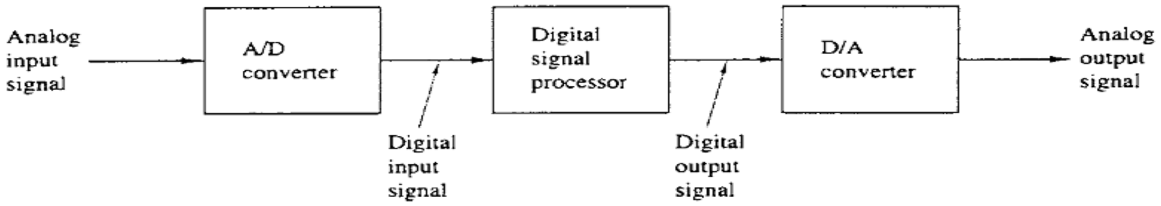


Fig 3. Block diagram of a DSP

Theoretical analysis and derivations are typically performed on discrete-time signal models, created by the abstract process of sampling. Numerical methods require a digital signal such as those produced by an analog-to-digital converter (ADC). The processed result might be a frequency spectrum or a set of statistics. But often it is another digital signal that is converted back to analog Formby a digital-to-analog converter (DAC). Even if that whole sequence is more complex than analog processing and has a discrete value range, the application of computational power to signal processing allows for many advantages over analog processing in many applications, such as error detection and correction in transmission as well as data compression.

CLASSIFICATION OF SIGNALS

Signals can be classified based on a lot of criteria, some of which are expressed below:

a. Based on nature of independent variable.
    i. Continuous time signal. ii. Discrete time signal
b. Based on the nature of the dependent variable or signal
    i. Analog signal ii. Digital signal.
c. Based on the number of independent variables
    i. One-dimensional signal. ii. Two-dimensional signal. iii. Multi-dimensional signal.
d. Based on the periodicity of the signal.
    i. Periodic signal ii. Aperiodic signal
e. Based on causality.
    i. Causal signal. ii. Anti-causal signal
f. Based on the energy content of the signal.
    i. Energy signal. ii. Power signal. iii. Neither power nor energy signal.
g. A signal could be Deterministic or Random.
h. A signal could be even or odd.
i. A signal could be Real, Imaginary or Complex.

In real life, most signals exist as analog signals in the sense that they are continuous and they have a value i.e. they have varying intensity at different points in time. For example, The intensity of light from the sun, noise in a market etc.

CONTINUOUS TIME SIGNALS

A continuous signal or a continuous-time signal is a varying quantity (a signal) whose domain, which is often time, is a continuum (e.g. a connected interval of the real signal). That is, the function's domain is an uncountable set. The function itself need not be continuous. Any analog signal is continuous by nature. Discrete-time signals, used in digital signal processing, can be obtained by sampling and quantization of continuous signals.

Continuous signal may also be defined over an independent variable other than time. Another very common independent variable is space and is particularly useful in image processing, where two space dimensions are used.

ANALOG SIGNAL

An analog signal has a theoretically infinite resolution. In practice an analog signal is subject to electronic noise and distortion introduced by communication channels and signal processing operations, which can progressively degrade the signal-to-noise ratio (SNR). In contrast, digital signals have a finite resolution. Converting an analog signal to digital form introduces a constant low-level noise called quantization noise into the signal which determines the noise floor, but once in digital form the signal can in general be processed or transmitted without introducing additional noise or distortion. In analog systems, it is difficult to detect when such degradation occurs. However, in digital systems, degradation can not only be detected but corrected as well.

DIGITAL SIGNAL

In digital signal processing, a digital signal is a representation of a physical signal that is a sampled and quantified. A digital signal is an abstraction which is discrete in time and amplitude. The signal's value only exists at regular time intervals, since only the values of the corresponding physical signal at those sampled moments are significant for further digital processing. The digital signal is a sequence of codes drawn from a finite set of values.[6] The digital signal may be stored, processed or transmitted physically as a pulse code modulation (PCM) signal.

COMPARISM	**ANALOG**	**DIGITAL**
Signal	Analog signal is a continuous signal which represents physical measurements.	Digital signals are discrete time signals generated by digital modulation.
Waves	Denoted by sine waves	Denoted by square waves
Representation	Uses continuous range of values to represent information	Uses discrete or discontinuous values to represent information
Examples	Human voice in air, analog electronic devices.	Computers, CDs, DVDs, and other digital electronic devices.
Technology	Analog technology records waveforms as they are.	Samples analog waveforms into a limited set of numbers and records them.
Data transmissions	Subjected to deterioration by noise during transmission and write/read	Can be noise-immune without deterioration during transmission and write/read cycle.

	cycle.	
Response to noise	More likely to get affected reducing accuracy	Less affected since noise response are analog in nature
Flexibility	Analog hardware is not flexible.	Digital hardware is flexible in implementation.
Uses	Can be used in analog devices only. Best suited for audio and video transmission.	Best suited for Computing and digital electronics.
Applications	Thermometer	PCs, PDAs
Bandwidth	Analog signal processing can be done in real time and consumes less bandwidth.	There is no guarantee that digital signal processing can be done in real time and consumes more bandwidth to carry out the same information.
Memory	Stored in the form of wave signal	Stored in the form of binary bit
Power	Analog instrument draws large power	Digital instrument draws only negligible power
Cost	Low cost and portable	Cost is high and not easily portable
Errors	Analog instruments usually have a scale which is cramped at lower end and give considerable observational errors.	Digital instruments are free from observational errors like parallax and approximation errors.

3.1 WHY DSP IS NEEDED

In the quest for increased performance, flexibility, configurability, communications, and remote monitoring and control, the power electronics industry is increasingly moving from analog to digital power converters, particularly when high-density power output is required.

Meeting all of these requirements often requires a custom solution from a power converter designer proficient in digital signal processing (DSP).

Digital signal processing (DSP) is also far more precise than analog. Settings do not drift with time and temperature changes since they are controlled only by the DSP clock and the software, not cap or resistor values that change over temperature and time.

The most important feature of a digital converter is its flexibility. Digital power converters offer an array of programmable parameters including output voltage settings, output current, current limit trip point, power sequencing routines, voltage margining, and multiple thresholds for warning and fault conditions for overcurrent, over temperature, and under- and overvoltage. Fault conditions and power usage can also be stored in non-volatile flash memory for later recall.

Designers can program any of these parameters at any point during the product's life cycle. These, and other function or feature changes, often simply require updating the flash memory, and can even be updated remotely over the internet. With analog, similar parameter or function changes require part (hardware) changes and often also require a new PC board.

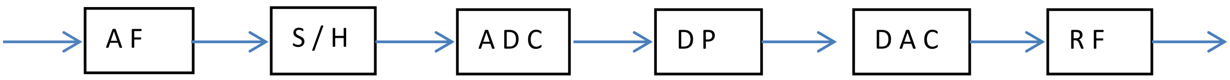
3.2 Why Asp Is Still In Used Though It Is An Analog Signal

Because sometimes analog signal processing is even cheaper, and can do things that digital processing cannot. Though Digital multiplication is simple and cheap, but if the original data and the required output are both analog, the cost and complexity of analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) to convert from analog to digital – and back again – often exceed those of an analog multiplier. Also, the digital propagation delay may be too great for a high-speed system.

In addition, it may be more efficient to process the analog signals before the ADC, even if digital data is required. An example is ac power measurement. If the signal to be measured is a simple 50 or 60 Hz sine wave and the load is resistive,

1. DIAGRAMMATIC SCHEME FOR DIGITAL PROCESSING OF ANALOG SIGNALS SHOWING THE BASIC OPERATIONS AND EXPLAINING HOW IT WORKS

- A F: Anti-aliasing filter
- S / H : Sample / Holding circuit
- A D C : Analog to Digital Converter
- D P : Digital Processor
- D A C : Digital to Analog Converter
- R F : Reconstruction Filter



Basic block diagram for digital processing of analog signals

ANTI-ALIASING FILTER: A filter used before a signal sampler, to restrict the bandwidth of the input analog signal to approximately satisfy the sampling theorem.

SAMPLE / HOLDING CIRCUIT: The sample and hold circuit is an analog device that samples (captures, grabs) the voltage of a continuously varying analog signal and holds (locks, freezes) its value at a constant level for a specified minimum period of time. Sample and hold circuits and related peak detectors are the elementary analog memory devices. They are typically used in analog-to-digital converters to eliminate variations in input signal that can corrupt the conversion process.

ANALOG TO DIGITAL CONVERTER: It's a device that converts a continuous physical quantity (in this case voltage) to a digital number that represents the quantity amplitude. The conversion involves quantization of the input, so it is necessarily introduces a small amount of error. Furthermore, instead of continuously performing the conversion, an ADC does the conversion periodically, sampling the input.

DIGITAL PROCESSOR: These are microprocessors specifically designed to handle Digital signal processing tasks. It requires a large number of mathematical operations to be performed quickly and repeatedly on a series of data samples. The analog signals which was converted to digital signal is then manipulated digitally and then converted back to analog signals. Most general-purpose microprocessors and operating systems can execute DSP algorithms successfully, but are not suitable for use in portable devices such as mobile phones and PDA because of power efficiency constraints.

DIGITAL TO ANALOG CONVERTER: It is a function that converts digital signals(usually binary) into an analog signal(current, voltage or electric charge). Unlike analog signals, digital signals can be transmitted, manipulated, and stored without degradation, albeit with more complex equipment. A DAC is needed to convert the digital signal to analog signals in other to drive an earphone or loudspeaker amplifier in other to produce sound (analog air pressure wave)

RECONSTRUCTION FILTER: It is a filter that smoothens the analog signal so that it can be interpretable(removing surrounding noise).

2. Advantages Of DSP

- They are capable of operating at a very wide operating range, require few external components, are easy to communicate with, and introduce a degree of flexibility in control.
- In DSP almost everything can be customized with just a little bit of code.
- DSP is very precise, the settings do not drift with time, and temperature changes, since they are controlled only by DSP clock and the software, not cap or resistor values that change over temperature and time.
- One important feature of DSP is its flexibility,
- Real time communications for monitoring and diagnostics, digital power devices can be tied into existing networked systems as well as communication processors and the information used to monitor and control the output.
- In oscilloscopes, DSP filtering ultimately can correct for hardware induced errors to improve measurement accuracy and enhanced display quality.
- Linear and nonlinear math operations work over a wide dynamic range of signal, 2^31 to 2^-31 for standard floating point. Also a suite of operations, like cos(), atan(), sqrt(), log() are available.
- Higher order filters can be implemented with a relatively low incremental cost. Additional memory and computations only.
- Filter design techniques provide a relatively high degree of freedom in spectral shaping, as in the Frequency Sampling method, for example.
- No tuning of analog components (R, L,C) during production or during maintenance.
- Good version control. Burn filter coefficients into memory and these will never change from one unit to the next.

- Software-based implementations require no custom hardware - just use standard signal I/O boards and write custom software.
- Small and rugged implementation using mixed-type VLSI, combining both DSP and analog I/O on a single chip.
- Adaptive filters become practical.
- Data compression becomes practical.

3. Advantages Of DSP Over ASP

- Digital signal processing is more complex in nature than analog signal processing; however it is has many advantages over ASP, such as error detection, correction in transmission, and data compression. Transtutors.com lists several advantages that DSP has over ASP:

- Digital signal processing operations can be changed by changing the program in digital programmable system, i.e., these is flexible systems.
- Better control of accuracy in digital systems compared to analog systems.
- Digital signals are easily stored on magnetic media such as magnetic tape without loss of quality of reproduction of signal.
- Digital signals can be processed off line, i.e., these are easily transported.
- Sophisticated signal processing algorithms can be implemented by DSP method.
- Digital circuits are less sensitive to tolerances of component values.
- Digital systems are independent of temperature, ageing and other external parameters.
- Digital circuits can be reproduced easily in large quantities at comparatively lower cost.
- Cost of processing per signal in DSP is reduced by time-sharing of given processor among a number of signals.
- Processor characteristics during processing, as in adaptive filters can be easily adjusted in digital implementation.
- Digital system can be cascaded without any loading problems.

4. Disadvantages Of DSP

- DSP techniques are limited to signals with relatively low bandwidths. Currently DSP devices are not faster enough to cater all the requirements in the world of signal processing. At the moment only up to moderate BW Signals can be processed. Bandwidths are in the 100MHz range are still being processed by analog methods since the DSP methods available are not faster enough.
- DSP Design time may be too long. And for DSP design, you should have a skilled, and perfect people who knows well about the DSP techniques. Also you should have the required toolset, specially the required software packages. Without these the DSP sometimes become impossible.
- The need for an ADC and DAC makes DSP uneconomical for simple applications(e.g. simple filters).
- Processing of signals involves more power consumption.
- Processing of signals beyond higher frequencies (beyond GHz) and below lower frequencies (a few Hz) involves limitations.
- When a piece of information is converted to digital form, some information is lost. Usually in most applications the minimum bit size are found given the maximum distortion that is presented to the designer. This follows from Rate Distortion Theory.
- Due to the same reason, when digital signal processing is done, due to floating point computation and limited accuracy, some information is lost. So architectures in which there is less of 'relevant' information is given priority, even if the number of computations is not minimum.

4.1 Digital Systems

Digital systems are hardware or software systems that process digital signals. Most digital systems are built up using binary or ‘‘on-off’’ logic, and so the operation of digital processors can be described using binary arithmetic. In contrast to the resistors, capacitors and inductors which makeup analog systems, the common building blocks for DSP systems are shift registers, adders and multipliers. These building blocks may be realized in either hardware or software, and if implemented in hardware, usually incorporate flip-flops, logic gates and synchronously controlled clocks. Like analog systems, discrete-time and digital systems can be analyzed in either the time or frequency domains.

A digital system is an interconnection of digital modules. To understand the operation of each digital module, it is necessary to have a basic knowledge of digital circuits and their logical function. A major trend in digital design methodology is the use of a HDL to describe and simulate the functionality of a digital circuit, it is important that students become familiar with an HDL-based design methodology

Properties of Digital Systems

A common name for modern discrete-time systems is digital systems . Digital systems have some important advantages in comparison with analog (continuous-time systems). They are more repeatable, less affected by environmental conditions, have better immunity to noise, and in most modern designs they are, at least to some extent, programmable. Being programmable means that a system's functions can be modified in software instead of in hardware. This is a great advantage because, generally speaking, changing software is much quicker and easier than changing hardware. Given the advantages and the rapid increase in capability accompanied by the rapid decrease in price at the same time, digital systems are very quickly becoming imbedded in every aspect of modern life.

Concept of Linearity Systems

Characterizing the complete input-output properties of a system by exhaustive measurement is usually impossible. When a system qualifies as a linear system, it is possible to use the responses to a small set of inputs to predict the response to any possible input. This can save the scientist enormous amounts of work, and makes it possible to characterize the system completely.

These notes explain the following ideas related to linear systems theory:

>>The challenge of characterizing a complex systems

>>Simple linear systems

*Homogeneity

*Superposition

>>Examples

*Stereo

*Swinging pendulum

Systems, Inputs, and Responses

Step one is to understand how to represent possible inputs to systems. Imagine a picture that shows the structure of the physical stimulus reaching your ear. On the horizontal axis we have time, and on the vertical axis we will plot the instantaneous density of the air molecules at your ear. Thus, we plot signal strength as a function of time. In the case of a simple hand-clap, the disturbance is a short, transient burst and is aptly named an impulse. It looks like a single upwards blip on the graph: the sound pressure momentarily increases when the clap hits your ear. More complex sounds look like more complex graphs on this kind of plot. This sort of graph offers a general way to describe all of the possible auditory stimuli.

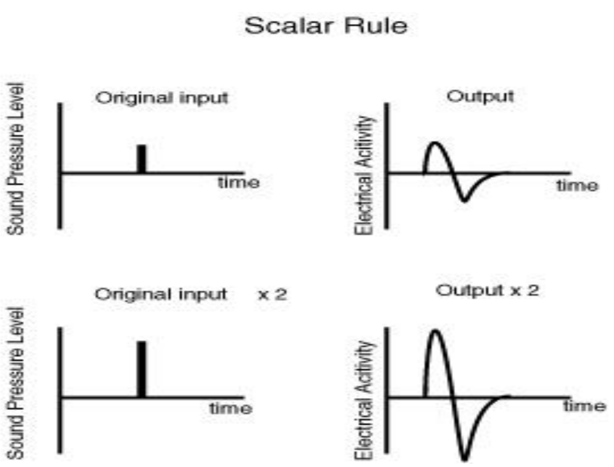
If we have a good theory about the kind of system we are studying, we can save a lot of time and energy by using the appropriate theory about the system's responsiveness. Linear systems theory is a good time-saving theory for linear systems which obey certain rules. Not all systems are linear, but many important ones are.

Linear Systems

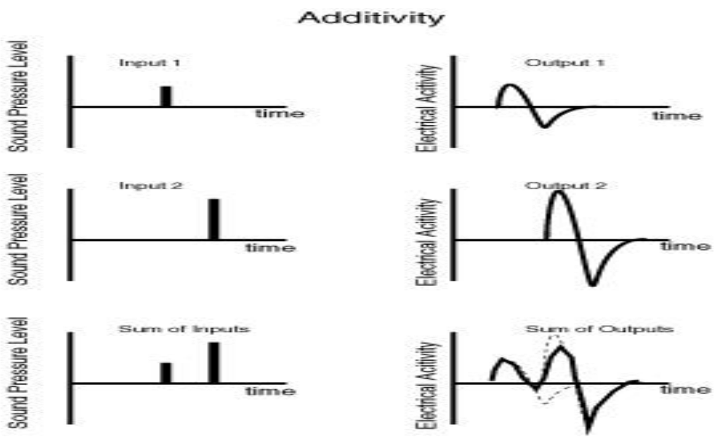
To see whether a system is linear, we need to test whether it obeys certain rules that all linear systems obey. The two basic tests of linearity are homogeneity and superposition.

Homogeneity: As we increase the strength of a simple input to a linear system, say we double it, then we predict that the output function will also be doubled. For example, if a person's voice becomes twice as loud, the ear should respond twice as much, if it's a linear system. This is called homogeneity or sometimes the scalar rule of linear systems.

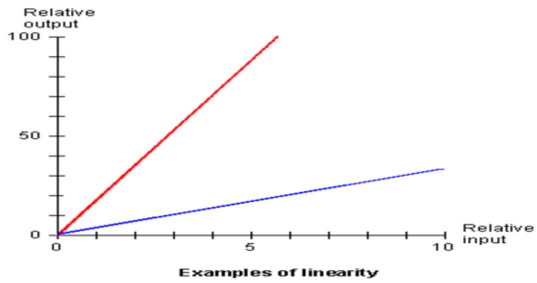
A system is said to be homogeneous if an amplitude change in the input results in an identical amplitude change in the output. That is if $x(n)$ results in $y(n)$ then, $kx(n)$ results in $ky(n)$ for any signal $x(n)$ and constant k.



Additivity: Suppose we present a complex stimulus S1 such as the sound of a person's voice to the inner ear, and we measure the electrical responses of several nerve fibers coming from the inner ear. Next, we present a second stimulus S2 that is a little different: a different person's voice. The second stimulus also generates a set of responses which we measure and write down. Then, we present the sum of the two stimuli S1 + S2: we present both voices together and see what happens. If the system is linear, then the measured response will be just the sum of its responses to each of the two stimuli presented separately.



Superposition: Systems that satisfy both homogeneity and additivity are considered to be linear systems. These two rules, taken together, are often referred to as the principle of superposition.



SUPERPOSITION PRINCIPLE WITH RESPECT TO D.S.P.

When we are dealing with linear systems, the only way signals can be combined is by *scaling* (multiplication of the signals by constants)followed by *addition*. For instance, a signal cannot be multiplied by another signal. This process of combining signals through scaling and addition is called **synthesis**.

Decomposition is the inverse operation of synthesis, where a single signal is broken into two or more additive components. This is more involved than synthesis, because there are infinite possible decompositions for any given signal. For example, the numbers 15 and 25 can only be synthesized (added) into the number 40. In comparison,

the number 40 can be decomposed into: 1 + 39 or 2 + 38 or -30.5 + 60 +10.5, etc.

Now we come to the heart of DSP: **superposition**, the overall strategy for understanding how signals and systems can be analysed.

Consider an input signal, called *x[n]*, passing through a linear system, resulting in an output signal, *y[n]*. The input signal can be decomposed into a group of simpler signals: *x0[n]*, *x1[n]*, *x2[n]*, etc. We will call these the **input signal components.** Next, each input signal component is individually passed through the system, resulting in a set of **output signal components**: *y0 [n]*, *y1 [n]*, *y2 [n]*, etc. These output signal components are then synthesized into the output signal, *y[n]*.

Here is the important part: the output signal obtained by this method is *identical* to the one produced by directly passing the input signal through the system. This is a very powerful idea. Instead of trying to understanding how *complicated* signals are changed by a system, all we need to know is how *simple* signals are modified. In the jargon of signal processing, the input and output signals are viewed as a *superposition* (sum) of simpler waveforms. This is the basis of nearly all signal processing techniques.

As a simple example of how superposition is used, multiply the number 2041 by the number 4, in your head. How did you do it? You might have imagined 2041 match sticks floating in your mind, quadrupled the mental image, and started counting. Much more likely, you used superposition to simplify the problem. The number 2041 can be decomposed into: 2000 + 40 + 1. Each of these components can be multiplied by 4 and then synthesized to find the final answer, i.e., 8000 + 160 + 4 = 8164.

CAUSAL AND NON CAUSAL SYSTEM

CAUSAL SYSTEMS

A system is said to be causal system if its output depends on present and past inputs only and not on future inputs.

A system which cannot recognize future is called a causal system. It is also called a realizable system. We cannot predict the future of a signal and use it for analysis in the system.

Examples: The output of casual system depends on present and past inputs, it means y(n) is a function of x(n), x(n-1), x(n-2), x(n-3)…etc. Some examples of causal systems are given below:

1) y (n) = x (n) + x (n-2)

2) y (n) = x (n-1) – x (n-3)

3) y (n) = 7x(n-5)

Significance of causal systems

Since causal system does not include future input samples; such system is practically realizable. That mean such system can be implemented practically. Generally all real time systems are causal systems; because in real time applications only present and past samples are present. Since future samples are not present; causal system is memory less system.

NON-CAUSAL SYSTEM

A system whose present response depends on future values of the inputs is called as a non-causal system. Non Causal signal processing is possible from recorded data. It is mostly applied in Geo-Physics and weather prediction. But a non-causal system cannot be realized in hardware. Only causal systems can be realized in hardware. Also called ANTI-CAUSAL SYSTEM

Examples: In this case, output y(n) is function of x(n), x(n-1), x(n-2)…etc. as well as it is function of x(n+1), x(n+2), x(n+3), … etc. following are some examples of non-causal systems:

1) Y (n) = x (n) + x (n+1)

2) Y (n) = 7x (n+2)

3) Y (n) = x (n) + 9x (n+5)

Significance of non-causal systems

Since non-causal system contains future samples; a non-causal system is practically not realizable. That means in practical cases it is not possible to implement a non-causal system.

- But if the signals are stored in the memory and at a later time they are used by a system then such signals are treated as advanced or future signal. Because such signals are already present, before the system has started its operation. In such cases it is possible to implement a non-causal system.
- Some practical examples of non-causal systems are as follows:

1) Population growth

2) Weather forecasting

3) Planning commission etc.

SOLVED PROBLEMS ON CAUSAL AND NON-CAUSAL SYSTEM

Determine if the systems described by the following equations are causal or non-causal.

1) y (n) = x (n) + x (n-3)

Solution: the given system is causal because its output (y (n)) depends only on the present x (n) and past x (n-3) inputs.

2) y (n) = x (-n+2)

Solution: this is non-causal system. This is because at n = -1 we get y (-1) = x [-(-1) +2] = x (3). Thus present output at n = -1, expects future input i.e. x(3)

<u>TIME INVARIANT</u>

A system is said to be Time Invariant if its input output characteristics do not change with time, otherwise it is said to be Time Variant system.

As already mentioned time invariant systems are those systems whose input output characteristics do not change with time shifting. Let us consider x(n) be the input to the system which produces output y(n) as shown.

Now delay input by k samples, it means our new input will become x (n-k). Now apply this delayed input x (n-k) to the same system as shown in figure below.

Now if the output of this system also delayed by k samples (i.e. if output is equal to y (n-k)) then this system is said to be Time invariant (or shift invariant) system.

If we observe carefully, x (n) is the initial input to the system which gives output y (n), if we delayed input by k samples output is also delayed by same (k) samples. Thus we can say that input output characteristics of the system do not change with time. Hence it is Time invariant system.

Theorem:

A system is Time Invariant if and only if

$$x(n) \xrightarrow{T} y(n) \text{ implies that } x(n-k) \rightarrow y(n-k)$$

Similarly a continuous time system is Time Invariant if and only if

$$x(t) \xrightarrow{T} y(t) \text{ implies that } x(t-k) \rightarrow y(t-k)$$

To determine whether the given system is Time Invariant or Time Variant, we have to follow the following steps:

Step 1: Delay the input x (n) by k samples i.e. x (n-k). Denote the corresponding output by y (n, k).

That means x (n-k) →y (n, k)

Step 2: In the given equation of system y (n) replace 'n' by 'n-k' throughout. Thus the output is y (n-k).

Step 3: If y (n, k) = y (n-k) then the system is time invariant (TIV) and if y (n, k) ≠ y (n-k) then system is time variant (TV).

Same steps are applicable for the continuous time systems.

Solved Problems:

1) Determine whether the following system is time invariant or not.

y (n) = x (n) – x (n-2)

Solution:

Step 1: Delay the input by 'k' samples and denote the output by y (n, k)

Therefore y (n, k) = x (n-k) – x (n-2-k)

Step 2: Replace 'n' by 'n-k' throughout the given equation.

Therefore y (n-k) = x (n-k) – x (n-k-2)

Step 3: Compare above two equations. Here y (n, k) = y (n-k). Thus the system is Time Invariant.

2) Determine whether the following systems are time invariant or not?

y (n) = x (n) + n x (n-2)

Solution:

Step 1: Delay the input by 'k' samples and denote the output by y (n, k)

Therefore y (n, k) = x (n-k) + n x (n-2)

Step 2: Replace 'n' by 'n-k' throughout the given equation.

Therefore y (n-k) = x (n-k) + (n-k) x (n-k-2)

Step 3: Compare above two equations. Here y (n, k) ≠ y (n-k). Thus the system is Time Variant.

5.1. OPERATION ON SEQUENCES

The following are the operations performed on sequences;

1. Sequence Addition
2. Scalar multiplication
3. Sequence Multiplication
4. Shifting
5. Reflection

1.1 **Sequence addition:**

Let {x[n]} and {y[n]}be two sequences. The sequence addition is defined as term by term addition. Let {z[n]} be the resulting sequence

{z[n]} = {x[n]} + {y[n]}

where each term z[n] = x[n] + y[n]

We will use the following notation

{x[n]} + {y[n]} = {x[n] + y[n]}

1.2 Scalar multiplication:

Let *a* be a scalar. We will take *a* to be real if we consider only the real valued signals, and take *a* to be a complex number if we are considering complex valued sequence. Unless otherwise stated we will consider complex valued sequences. Let the resulting sequence be denoted by {w[n]}

{w[n]} = a {x[n]}

is defined by

w[n] = ax[n]

each term is multiplied by *a*

We will use the notation

a {w[n]} = {aw[n]}

Note: If we take the set of all sequences and define these two operations as addition and scalar multiplication they satisfy all the properties of a linear vector space.

.

Sequence multiplication:

Let {x[n]} and {y[n]} be two sequences, and {z[n]} be resulting sequence

{z[n]} = {x[n]}{y[n]}

where z[n] = x[n] y[n]

The notation used for this will be {x[n]} {y[n]} = {x[n] y[n]}

Shifting:

This is also known as translation. Let us shift a sequence {x[n]} by n_0 units, and the resulting sequence be {y[n]}

$$\{y[n]\} = z^{-n_0}(\{x[n]\})$$

where $z^{-n_0}()$ is the operation of shifting the sequence right by n_0 unit. The terms are defined by $y[n] = x[n - n_0]$. We will use short notation $\{x[n - n_0]\}$ to denote shift by n_0.

Figure below show some examples of shifting.

{x[n]} {x[n-2]} {x [n+1]}

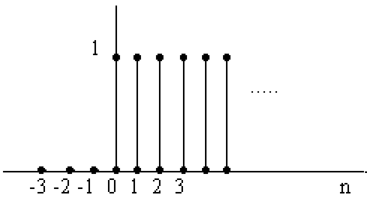


Fig 1.0

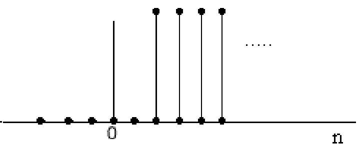


Fig 1.1

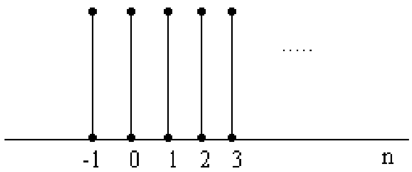


Fig 1.2

A negative value of no (fig 1.1) means shift towards the right (Fig 1.2), while a positive value on no means shift towards left (Fig 1.3).

Reflection:

Let {x[n]} be the original sequence, and {y[n]} be reflected sequence, then y[n] is defined by

y[n] = x[-n]

{x[n]}

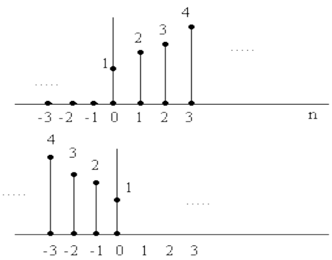


Fig 1.3

2. DIFFERENCE EQUATION

A Difference Equation is an equation relating the past and present samples of the output signal with past and present samples of the input signal. It is also called a recursion equation and filters that use it are called recursive filters. The difference equation describes the recursive filter.

$$a_0 \mathrm{y[n]} = \mathrm{b_0 x[n] + b_1 x[n-1] + b_2 x[n-2]} + \cdots - \mathrm{a_1 y[n-1] - a_2 y[n-2] - a_3 y[n-3]} - \cdots$$

Where x [ ] and y [ ] are the input and output signals respectively, and the 'a' and 'b' terms are the recursion coefficients with $a_0 = 1$.

While a difference equation describes a discrete system where old values from the system are used to calculate new values, a differential equation describes a continuous system where the rates of change are defined in terms of other values in the system.

A use of this equation is to describe how a programmer would implement the filter. It also finds application in population dynamics. An equally important aspect is that it represents a mathematical relationship between the input and output that must be continually satisfied.

Linear Time Invariant Difference Equation

The recursive difference equation form is given by

$$y[n] = -\sum_{k=1}^{N} a_k y[n-k] + \sum_{k=0}^{M} b_k x[n-k]$$

Re-written as

$$y[n] + \sum_{k=1}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k]$$

This is the ordinary linear time invariant (LTI) difference equation of nth order. By taking the z-transform of each term on both sides using the properties of linearity and time-shifting, we obtain,

$$Y(z) + Y(z)\sum_{k=1}^{N} a_k z^{-k} = X(z)\sum_{k=0}^{M} b_k z^{-k}$$

Thus

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

This can be further expressed in a factored form

$$H(z) = b_0 \left[\frac{\prod_{k=1}^{M}(1 - c_k z^{-1})}{\prod_{k=1}^{N}(1 - d_k z^{-1})}\right]$$

Each of the factors $(1 - c_k z^{-1})$ in the numerator contributes a zero at $z = c_k$ and a pole at $z = 0$. Similarly, each of the factors $(1 - d_k z^{-1})$ in the denominator contributes a pole at $z = d_k$ and a zero at $z = 0$. The poles determine the stability.

3. ALIASING

In signal processing and related disciplines, aliasing is an effect that causes different signals to become indistinguishable (or aliases of one another) when sampled. It also refers to the distortion or artifact that results when the signals reconstructed from samples is different from the original continuous signal.

Aliasing can occur in signals sampled in time, for instance digital audio, and is referred to as temporal aliasing. Aliasing can also occur in spatially sampled signals, for instance digital images. Aliasing in spatially sampled signals is called spatial aliasing. Aliasing can also occur whenever the use of discrete elements to capture or produce a continuous signal causes frequency ambiguity.

In sound and image generation, aliasing is the generation of a false(alias) frequency along with the correct one when doing frequency sampling. For images, this produces a jagged edge, or stair-step effect. For sound, it produces a buzz. An example of spatial aliasing is the moiré pattern one can observe in a poorly pixelized image of a brick wall. Spatial anti-aliasing techniques avoid such poor pixelizations. Aliasing can be caused either by the sampling stage or the reconstruction stage; these may be distinguished by calling sampling aliasing preleasing and reconstruction aliasing post aliasing.

Temporal aliasing is a major concern in the sampling of video and audio signals. Music, for instance, may contain high-frequency components that are inaudible to humans. In video or cinematography, temporal aliasing results from the limited frame rate, and causes the wagon-wheel effect, whereby a spoked wheel appears to rotate too slowly or even backwards. Aliasing has changed its apparent frequency of rotation. Temporal aliasing frequencies in video and cinematography are determined by the frame rate of the camera, but the relative intensity of the aliased frequencies is determined by the shutter timing(exposure time) or the use of a temporal aliasing reduction filter during filming.

BANDLIMITED FUNCTIONS

Actual signals have finite duration and their frequency content, as defined by the Fourier transform, has no upper bound. Some amount of aliasing always occurs when such functions are sampled. Functions whose frequency content is bounded(band limited) have infinite duration. If sampled at a high enough rate, determined by the bandwidth, the original function can in theory be perfectly reconstructed from the infinite set of samples.

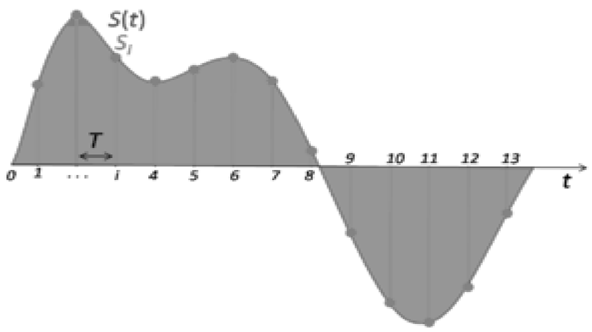
BANDPASS SINALS

Sometimes aliasing is used intentionally on signals with no low-frequency content, called bandpass signals. Under sampling, which creates low-frequency aliases, can produce the same result, with less effort, as frequency-shifting the signal to lower frequencies before sampling at the lower rate. Some digital channelizers exploit aliasing in this way for computational efficiency.

4. SAMPLING: OVER SAMPLING AND UNDER SAMPLING

Broadly, we have two (2) types of signal; continuous time signal and discrete time signal. Although in reality we have a large number of continuous time signal, but there is preference for discrete time signal due to some recent advance development in digital technology over the years and the more flexibility of discrete time signal i.e. easily programmable. This pose the problem of converting continuous time signal to discrete time signal, this is solved by the phenomenon known as *Sampling.*

We utilize sampling to convert a continuous time signal to a discrete time signal, process the discrete time signal using a discrete time system and then convert back to continuous time signal.



The continuous signal is represented with the curved line while the discrete samples are indicated by the vertical lines.

Fig4.0 Signal sampling representation.

OVERSAMPLING

In signal processing, oversampling is the process of sampling a signal with a sampling frequency significantly higher than the nyquist rate. It improves resolution, reduces noise and helps to avoid aliasing and phase distortion.

UNDERSAMPLING

In signal processing, under sampling is a process of sampling a signal with a sampling frequency below its nyquist rate

Nyquist rate: This is twice the maximum frequency of the signal.

5. SAMPLING THEOREM

The signals we use in the real world, such as our voices, are called "analog" signals. To process these signals in computers, we need to convert the signals to "digital" form. While an analog signal is continuous in both time and amplitude, a digital signal is discrete in both time and amplitude. To convert a signal from continuous time to discrete time, a process called sampling is used. The value of the signal is measured at certain intervals in time. Each measurement is referred to as a sample. (The analog signal is also quantized in amplitude, but that process is ignored in this demonstration. See the Analog to Digital Conversion page for more on that.)

When the continuous analog signal is sampled at a frequency F, the resulting discrete signal has more frequency components than did the analog signal. To be precise, the frequency components of the analog signal are repeated at the sample rate. That is, in the discrete frequency response they are seen at their original position, and are also seen centered around +/- F, and around +/- 2F, etc.

How many samples are necessary to ensure we are preserving the information contained in the signal? If the signal contains high frequency components, we will need to sample at a higher rate to avoid losing information that is in the signal. In general, to preserve the full information in the signal, it is necessary to sample at twice the maximum frequency of the signal. This is known as the Nyquist rate. The Sampling Theorem states that a signal can be exactly reproduced if it is sampled at a frequency F, where F is greater than twice the maximum frequency in the signal.

What happens if we sample the signal at a frequency that is lower that the Nyquist rate? When the signal is converted back into a continuous time signal, it will exhibit a phenomenon called aliasing. Aliasing is the presence of unwanted components in the reconstructed signal. These components were not present when the original signal was sampled. In addition, some of the frequencies in the original signal may be lost in the reconstructed signal. Aliasing occurs because signal frequencies can overlap if the sampling frequency is too low. Frequencies "fold" around half the sampling frequency - which is why this frequency is often referred to as the folding frequency.

Sometimes the highest frequency components of a signal are simply noise, or do not contain useful information. To prevent aliasing of these frequencies, we can filter out these components before sampling the signal. Because we are filtering out high frequency components and letting lower frequency components through, this is known as low-pass filtering.

6. RECURSIVE AND NON-RECURSIVE

Digital filters can be classified based on filter structure as recursive and non-recursive filters. In general the output of a function can be based on future, current and past input values, as well as past output values. A filter in which its output is a function of its past outputs is known as a **RECURSIVE FILTER**. A recursive filter can be recognized from the equations below, since at least one a_k coefficient is non-zero, for $1 \leq k \leq M$, and at least one b_k coefficient, for $1 \leq k \leq M$ is non-zero.

$$y(n) = \sum_{k=1}^{M} a_k y(n-k) + \sum_{k=-N_p}^{N_p} b_k \, x(n-k) \qquad \text{eqn. 6.0}$$

$$y(n) = \sum_{k=1}^{M} b_k z^{-k} \,/\, \left(1 - \sum_{k=-N_p}^{N_p} a_k z^{-k}\right) \qquad \text{eqn. 6.1}$$

If the filter output value is a function of only input sequence values, it is called a **NON-RECURSIVE FILTER**. A recursive filter is easily recognizable from the above defining equations, for which $a_k = 0$ or

$b_k = 0$ for all k. All poles of non-recursive filter is at either z = 0 or z = ∞. It can also be called an all zero filter.

Using the following elements to describe a digital filter,

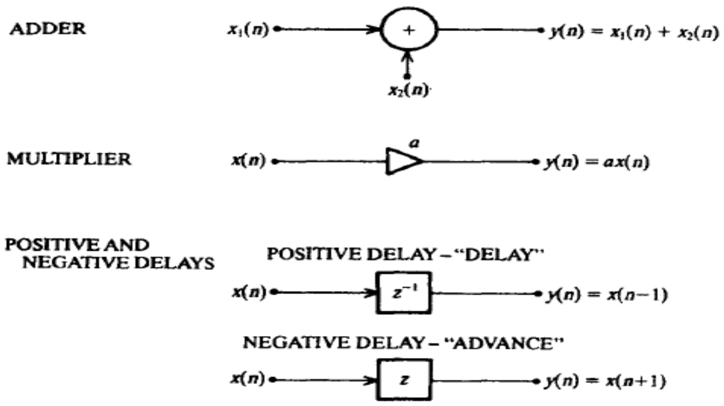


fig. 6.0

We can describe a first order non-recursive filter given by

$$y(n) = ay(n-1) + x(n) \qquad \text{eqn. 6.2}$$

The fig 1.1 shows a first-order recursive filter.

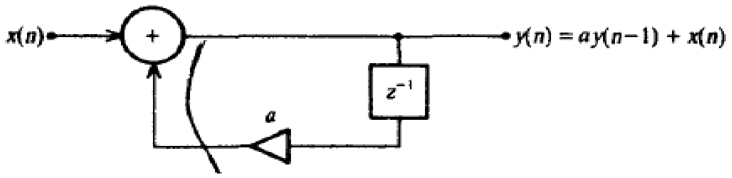


fig. 6.1

An example of a non-recursive filter is described by the equation

$$y(n) = \frac{1}{3}x(n+1) + \frac{1}{3}x(n) + \frac{1}{3}x(n-1) \text{eqn. 6.3}$$

And can be visualized from fig. 1.2 below,

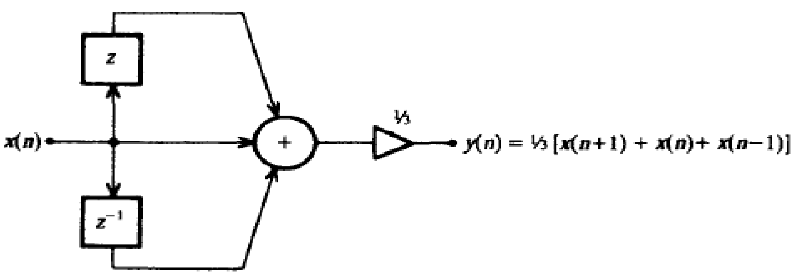


fig. 6.2

6.0FILTERS

1.0 Introduction: In signal processing, a filter is a device or process that removes some unwanted components or features from a signal. Filtering is a class of signal processing, the defining feature of filters being the complete or partial suppression of some aspect of the signal. Most often, this means removing some frequencies and not others in order to suppress interfering signals and reduce background noise. However, filters do not exclusively act in the frequency domain; especially in the field of image processing many other targets for filtering exist.

Electronic filters are circuits which perform signal processing functions, specifically to remove unwanted frequency components from the signal, to enhance wanted ones, or both. A filter can then be said as a circuit designed to perform frequency selection. A common need for filter circuits is in high – performance stereo systems where certain ranges of audio frequencies need to be amplified or suppressed for best sound quality and power efficiency.

There are many different bases of classifying filters and those overlap in many different ways; there is no simple hierarchical classification. Filters may be:

(i) Linear or non-linear

(ii) time-invariant or time-variant, also known as shift invariance. If the filter operates in a spatial domain then the characterization is a space invariance.

(iii) casual or not-casual: depending if present output depends or not on future input; of course, for time related signals processed in real time all the filters are casual; it is not necessarily so for filters acting on space related signals or for deferred time processing of time related signals.

(iv) Analog or digital

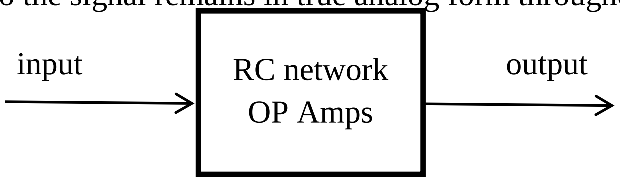
(v) discrete-time (sampled) or continuous-time

(vi) Passive or active type of continuous-time filter

(vii) Infinite impulse response (IIR) or finite impulse response (FIR) type of discrete-time or digital filter.

2.0 Types of filters

2.1 Analog filters: They are the basic building block of signal processing much used in electronics . They are much simpler filters because they don't have to convert signal to a digital form before processing, so the signal remains in true analog form throughout the process.



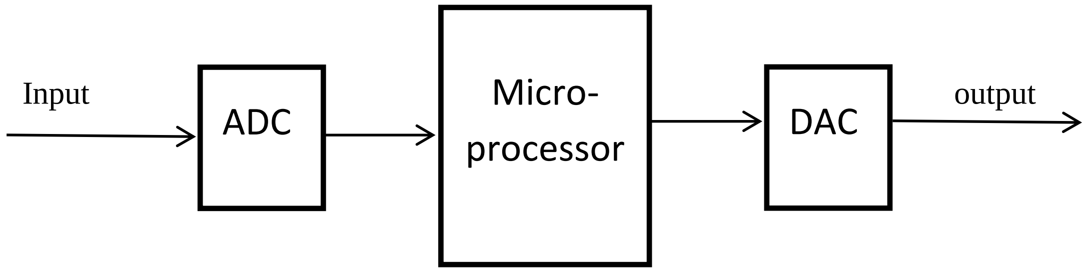
RC networks which are made up of simple resistor and capacitor components, do the filtering. There is also an op-amp involved to amplify the signal to a required level.

An analog filter takes the analog signal as input and process the signal and finally gives the analog output.

2.1.0 Merits of analog filter

1. They are simple to implement- Implementation is straight forward
2. There is no requirement for a software programmer to write the algorithm.
3. Simple RC filters require minimal components and are sensitive.
4. They are cheap and have a large dynamic range in both amplitude and frequency.

2.2 Digital filters: In signal processing, a digital filter is a system that performs mathematical operations on a sampled discrete-time signal to reduce or enhance certain aspects of that signal.



It consists of an ADC that converts the analog voltage levels in the waveform into binary numbers, Followed by a micro-processor which processes the binary numbers. The program instruction (software) running on the micro-processor implements the digital filter by performing the necessary mathematical numbers received from the ADC. [an algorithm embedded into the micro-processor and it determines the filtering characteristics]. Once the binary number are processed, or mathematically manipulated, they are fed back to another circuit called a DAC, thereby reconstructing the signal back into an analog form. A digital filter consists of elements like adder, multiplier and delay units. They are vastly superior in the level of performance in comparison to analog filters.

2.2.1 Merits of digital filters

1. Unlike analog filters the digital filter performance is not influenced by component ageing, temperature and power variations.
2. A digital filter is highly immune to noise and is relatively stable.
3. Digital filters can handle low frequency signals accurately.
4. It can be easily designed, tested and implemented on a general-purpose computer or workstation and multiple filtering is possible.
5. It can be change be adjusted or changed without affecting the circuitry. This is because it’s programmable.
6. It has repeatable performance unit to unit.
7. It is possible to filter several input sequences without any hardware replication.
8. All data can be stored

2.2.2 Demerits of digital filter

1. They are relatively expensive and quit technical to implement.
2. In reality, the signal bandwidth of the digital sequence is much lower than the analog sequence.
3. Signal processing speed is one of the key factors of calculating the total device performance. Actually the speed operation totally depends on the number of the arithmetic operation in the processor.
4. Finite word-length effect, which results quantizing noise and round-off noise, is another major drawback during computation.
5. It needs much longer time to design and develop the digital sequences though it can be used on other tasks or applications once developed.

2.3Impulse filters: There are two fundamental types of impulse filters, finite impulse response filter and Infinite impulse response filters.

2.3.1finite impulse response: The finite impulse response filter indicates a filter whose response depends only on the present and past input samples. So in finite impulse response, the impulse response sequence is of finite duration, i.e. it has a finite number of non-zero terms.

2.3.1.1 Merits of finite impulse response filters

1. FIR filters are stable.
2. FIR filters can achieve performance levels that are not possible with analog filter techniques (such as perfect linear phase response).
3. FIR filters reduces the computation complexity.
4. Round off noise can be eliminated in FIR filters.
5. FIR filters can be designed with exact linear phase. These linear phase filters are important for applications where frequency dispersion due to non-linear phase is hazardous (e.g. speech processing and data transmission).
6. FIR filters can be efficiently implemented in multirate DSP systems – making implementation simpler.
7. It requires no feedback.

2.3.1.2 Demerits of finite impulse response filters

1. It requires more computation power.
2. It generally requires a large number of multiply accumulators and therefore requires fast and efficient DSPs.
3. As large number of impulse response samples are required to properly approximate sharp cut off FIR filters, the processing will become complex due to convolution.
4. The delay of linear phase FIR filters can sometimes create problems in some DSP applications.

2.3.2 Infinite impulse response (IIR) filters

Infinite impulse response filters get their name because their impulse response extends for an infinite period of time. This is because they are recursive, i.e., they utilize feedback. Although they can be implemented with fewer computations than FIR filters, IIR filters are generally implemented in 2-pole sections called biquads because they are described with a biquadratic equation in the Z-domain.

The general form of the IIR equations contains the general transfer function H(z), which contains polynomials in both the numerator and the denominator. The roots of the denominator determine the pole locations of the filter, and the roots determine the zero locations.

The reduced form of the IIR equation is given as

$$y(n) = b_0x(n) + b_1x(n-1) + b_2x(n-2) - a_1y(n-1) - a_2y(n-2)$$

2.3.2.1 Merits of infinite impulse response filter

1. An IIR filter has lesser number of side lobes in the step band than an FIR filter unit with the same number of parameters.
2. Implementation of an IIR filter involves less parameters, less memory requirement and lower computational complexity.

2.3.2.2 Demerits of infinite impulse response filter

1. They do not have linear phase and also they are not stable.
2. Realization of IIR filters is not very easy as compared to FIR filters.

3. Because of its recursive property, the number of coefficients is very large and the memory requirements are also high.

3.0 Order of a filter

It's the largest number of previous inputs or output values required to compute the current output.

Zero order => $y(n) = a_0\, x_n$

First order => $y(n) = a_0\, x_n + a_1\, x_{n-1}$

Second order => $y(n) = a_0\, x_n + a_1\, x_{n-1} + a_2\, x_{n-2}$

$y(n) = b_0 x(n) + b_1 x(n-1) + \ldots\ldots\ldots + b_n x(n-N)$

where x(n) = input signal, y(n) = output signal, N = Filter order, an Nth order filter has (N+1) terms in RHS. bi = value of the response at the ith instant.

4.0 Digital filter coefficient

The constants a_0, a_1, a_2, a_3,……… appearing in these expressions above are called the filter coefficients, the values of these coefficients determine the characteristics of a particular filter.

5.0 The unit delay operator [z^{-1}]

When applied to a sequence of digital values, the operator gives the previous value in the sequence. It therefore introduces a delay of one scanning interval.

When you apply the operator Z^{-1} to an input value, say (x_n) it gives the previous input (x_{n-1}).

Therefore if $x_o = 5$, then $Z^{-1}\, x_1 = x_o = x_s$

It can also be applied more than once.

$Z^{-1}(Z^{-1}\, x_n) = Z^{-1}\, x_{n-1} = x_{n-2}$

6.0 Uses of filters

Filters have two uses

I. SIGNAL SEPARATION:- Needed when a signal has been contaminated with interference, noise or other signals. For example, imagine a device for measuring the electrical activity of a baby's heart while still in the womb. The raw signal will likely be corrupted by the breathing and heartbeat of the mother. A filter might be used to separate these signals so that they can be more easily analyzed.

II. SIGNAL RESTORATION:- it is used when a signal has been distorted in same way some way or the other. Example an audio recording made with poor equipment may be filtered to better represent the sound as it actually occurred. Another example is DE blurring of an image acquired with an improperly focused camera lens or a shaky camera.
These problems can be attacked with either an analog or digital filters.