GaussianWaves
Signal Processing Simplified

≡ Menu

Choosing FIR or IIR ? Understand design perspective

February 17, 2017 by Mathuranathan

"What is the best filter that I should use? FIR or IIR ?" is often the question asked by many. There exists two different types of Linear Time Invariant (LTI) filters from transfer function standpoint : FIR (Finite Impulse Response) and IIR (Infinite Impulse Response) filters and myriad design techniques for designing them. The mere fact that there exists so many techniques for designing a filter, suggests that there is no single optimal filter design. One has to weigh-in the pros and cons of choosing a filter design by considering the factors discussed here.

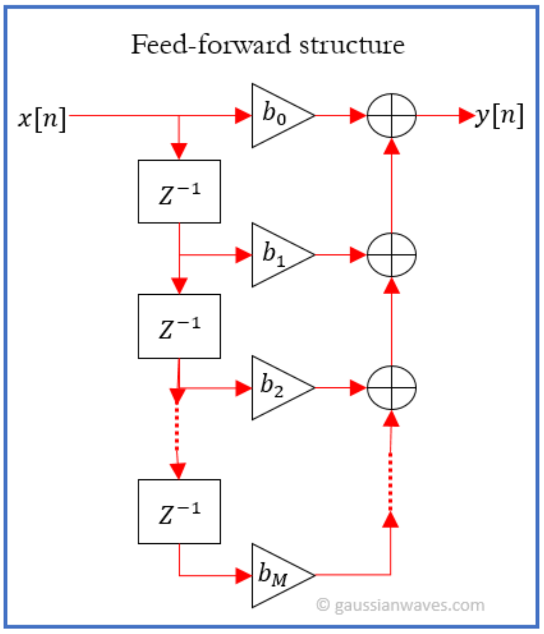
Before proceeding to know the secrets of choosing a filter, I urge you to brush up the fundamentals of digital filter design.

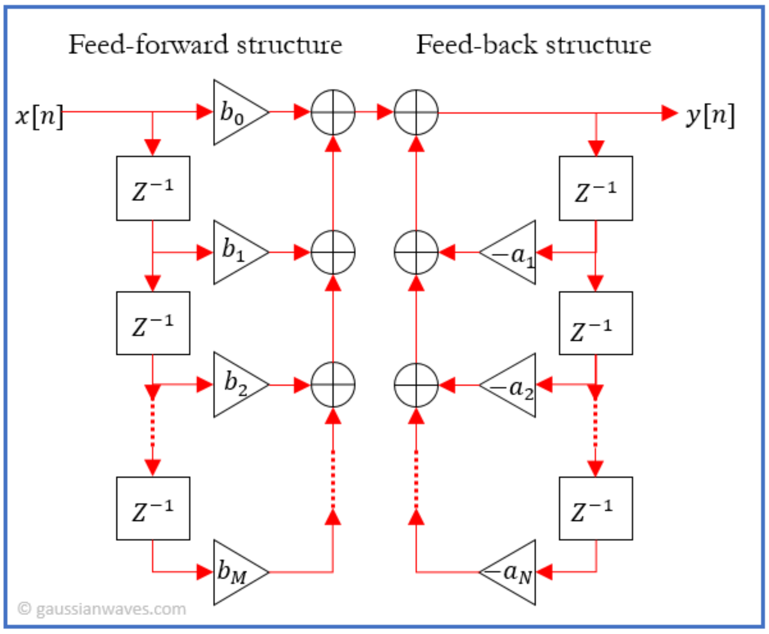
Design specification:

A filter design starts with a specification. We may have a simple specification that just calls for removing an unwanted frequency component or it can be a complete design specification that calls for various parameters like – amount of ripples allowed in passband, stop band attenuation, transition width etc.., The design specification usually calls for satisfying one or more of the following:

- Desired Magnitude response – $|H_{spec}(\omega)|$
- Desired Phase response – $\angle H_{spec}(\omega)$
- Tolerance specifications – that specifies how much the filter response

allowed to vary when compared with ideal response. Examples include how much ripples allowed in passband, stop band etc..,





(a) (b)

Figure 1: FIR and IIR filters can be realized as direct-form structures. (a) Structure with feed-forward elements only – typical for FIR designs. (b) Structure with feed-back paths – generally results in IIR filters. Other structures like lattice implementations are also available.

Given the specification above, the goal of a filter design process to choose the parameters M, N, b_k and a_k, such that the ***transfer function*** of the filter

$$H(z) = \frac{\sum\limits_{i=0}^{M} b_k z^{-1}}{1 + \sum\limits_{i=1}^{N} a_k z^{-1}} = \frac{\prod\limits_{i} (z - z_i)}{\prod\limits_{j} (z - p_j)} \qquad (1)$$

yields the desired response: $H(\omega) \approx H_{spec}(\omega)$. In other words, the design process also involves choosing the number and location of the zeros zeros (z_i) and poles (p_j) in the pole-zero plot.

Two types of filter can manifest from the given transfer function above.

- When $N = 0$, there is no feedback in the filter structure, no poles in the pole-zero plot (in fact all the poles sit at the origin). The impulse respon

of such filter dies out (becomes zero) beyond certain point of time and it is classified as Finite Impulse Response (FIR) filter. It provides linear phase characteristic in the passband.
• When $N > 1$, the filter structure is characterized by the presence of feedback elements. Due to the presence of feedback elements, the impulse response of the filter may not become zero beyond certain point, but continues indefinitely and hence the name Infinite Impulse Response (IIR) filter.
• Caution: In most cases, the presence of feedback elements provide infinite impulse response. It is not always true. There are some exceptional cases where the presence of feedback structure may result in finite impulse response. For example, a moving average filter will have a finite impulse response. The output of a moving average filter can be described using a recursive formula, which will result in a structure with feedback elements.

General considerations in design:

As specified earlier, the choice of filter and the design process depends on design specification, application and the performance issues associates with them. However, the following general considerations are applied in practical design.

Minimizing number of computations

In order to minimize memory requirements for storing the filter co-efficients $\{a_k\}$, $\{b_k\}$ and to minimize the number of computations, ideally we would like $N + M$ to be as small as possible. For the same specification, IIR filters result in much lower order when compared to its FIR counter part. Therefore, IIR filters are efficient when viewed from this standpoint.

Need for real-time processing

The given application may require processing of input samples in real-time or the input samples may exist in a recorded state (example: video/audio playback, image processing applications, audio compression). From this perspective, we have two types of filter systems

• **Causal Filter**
— Filter output depends on present and past input samples, not on the future samples. The output may also depend on the past output sample

in IIR filters. Strictly no future samples.
— Such filters are very much suited for real-time applications.

• **Non-Causal Filter**
— There are many practical cases where a non-causal filter is required. Typically, such application warrants some form of post-processing, where the entire data stream is already stored in memory.
— In such cases, a filter can be designed that can take in all type of input samples : present, past and future, for processing. These filters are classified as non-causal filters.
— Non-causal filters have much simpler design methods.

It can be often seen in many signal processing texts, that the causal filters are practically realizable. That does not mean non-causal filters are not practically implementable. In fact both types of filters are implementable and you can see them in many systems today. The question you must ask is : whether your application requires real-time processing or processing of pre-recorded samples. If the application requires ***real-time processing***, causal filters must be used. Otherwise, non-causal filters can be used.

Consequences of causal filter:

If the application requires real-time processing, causal filters are the only choice for implementation. Following consequences must be considered if causality is desired.

Ideal filters with finite bands of zero response (example: brick-wall filters), cannot be implemented using causal filter structure. A direct consequence of causal filter is that the response cannot be ideal. Hence, we must design the filter that provides a close approximation to the desired response . If tolerance specification is given, it has to be met.

For example, in the case of designing a low pass filter with given passband frequency (ω_P) and stopband frequencies (ω_S), additional tolerance specifications like allowable passband ripple factor (δ_P), stopband ripple factor ($\delta_S$latex) need to be considered for the design,. Therefore, the practical filter design involves choosing $\omega_P, \omega_S, \delta_P$ and δ_S and then designing the filter with $N, M, \{a_k\}$ and $\{b_k\}$ that satisfies all the given requirements/responses. Often, iterative procedures may be required to satisfy all the above (example: Parks and McClellan algorithm used for designing optimal causal FIR filters ***[1]***).

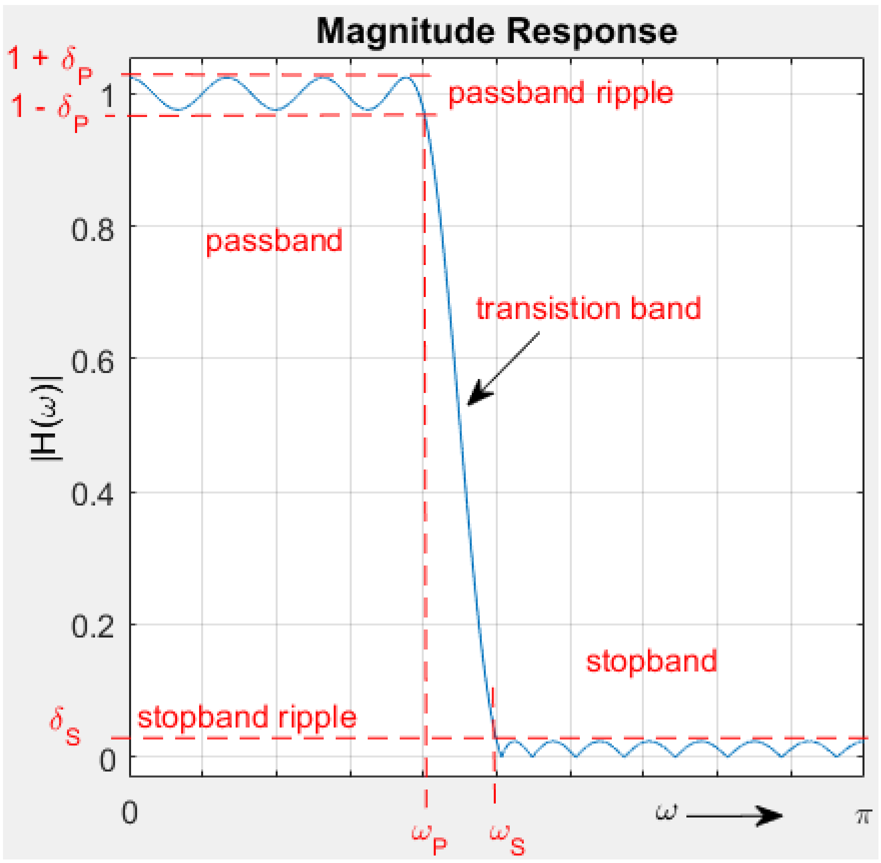


Figure 2: A sample filter design specification

For a causal filter, frequency response's real part $H_R(\omega)$ and the imaginary part $H_I(\omega)$ become Hilbert transform pair [2]. Hence the magnitude and phase responses become interdependent.

Stability

For a causal LTI digital filter will be BIBO (Bounded Input Bounded Output) stable, if and only if the impulse response $h[n]$ is absolutely summable.

$$\sum_{n=-\infty}^{n=\infty} |h[n]| < \infty \qquad (2)$$

Impulse response of ***FIR filters are always bounded and hence they are inherently stable***. On the other hand, an ***IIR filter may become unstable if not designed properly***.

Consider an IIR filter implemented using a floating point processor that has enough accuracy to represent all the coefficients in the transfer function below

$$H_1(z) = \frac{1}{1 - 1.845\ z^{-1} + 0.850586\ z^{-2}} \qquad (3)$$

The corresponding impulse response $h_1[n]$ is plotted in Figure 3(a). The plot shows that the impulse response decays rapidly to zero as n increases. For this case, the sum in equation (2) will be finite. Hence this IIR filter is stable.

Suppose, if we were to implement the same filter in a fixed point processor and we are forced to round-off the co-efficients to 2 digits after the decimal point, the same transfer function looks like this

$$H_2(z) = \frac{1}{1 - 1.85\ z^{-1} + 0.85\ z^{-2}} \qquad (4)$$

The corresponding impulse response $h_2[n]$ plotted in Figure 3(b) shows that the impulse response increases rapidly towards a constant value as n increases. For this case, the sum in equation (2) will tend to infinity. Hence the implemented IIR filter is unstable.

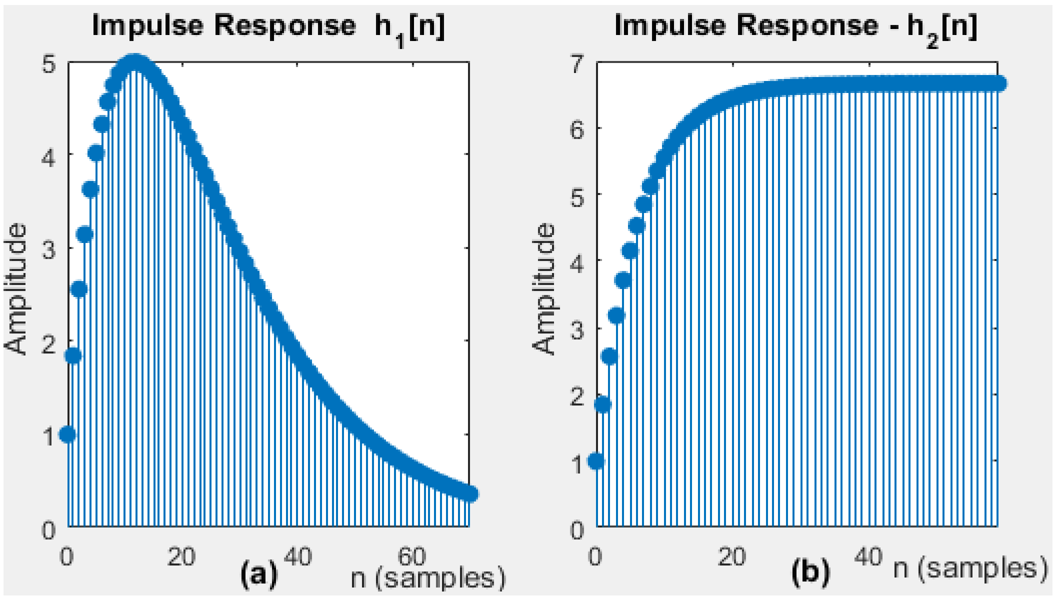


Figure 3: Impact of poorly implemented IIR filter on stability. (a) Stable IIR filter, (b) The same IIR filter becomes unstable due to rounding effects.

Therefore, it is imperative that an IIR filter implementation need to be tested for stability. To analyze the stability of the filter, the infinite sum in equation *(2)* need to be computed and it is often difficult to compute thi sum. ***Analysis of pole-zero plot*** is an alternate solution for this problem. Tc have a stable causal filter, the poles of the transfer function should lie completely ***strictly inside*** the unit circle on the pole-zero plot. The pole-zero plot for the above given transfer functions $H_1(z)$,$H_2(z)$ are plotted in Figure 4. It shows that for the transfer function $H_1(z)$, all the poles lie within the unit circle (the region of stability) and hence it is a stable IIR filter. On the other hand, for the transfer function $H_2(z)$, one poles lie exactly on the unit circle (ie, it is just out of the region of stability) and hence it is an unstable IIR filter.

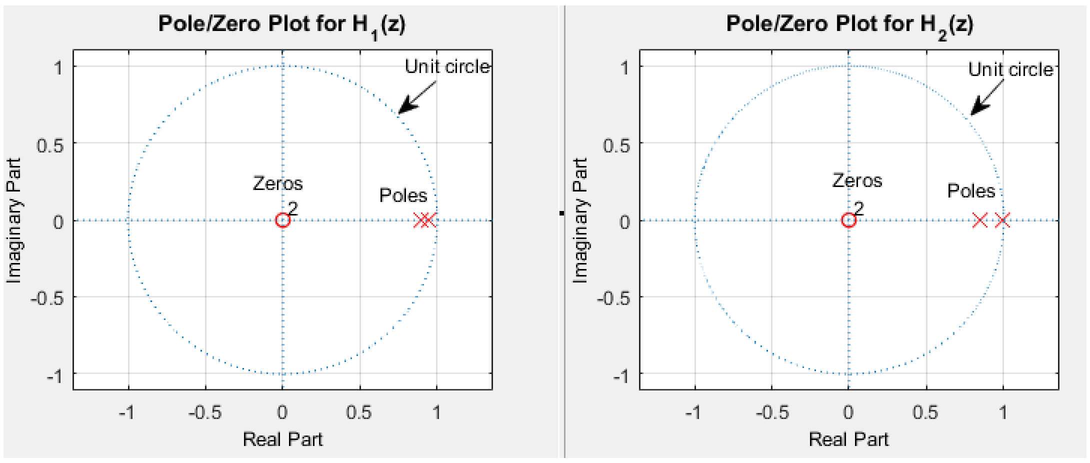


Figure 4: Impact of poorly implemented IIR filter on stability. (a) Stable IIR filter since all poles lie inside the unit circle, (b) Due to rounding effects, the one pole lies on the unit circle (just outside the region of stability), hence it becomes unstable.

Linear phase requirement

In many signal processing applications, it is needed that a digital filter should not alter the angular relationship between the real and imaginary components of a signal, especially in the passband. In otherwords, the phase relationship between the signal components should be preserved in the filter's passband. If not, we have ***phase distortion***.

Phase distortion is a concern in many signal processing applications. For example, in phase modulations like GMSK [3],the entire demodulation process hinges on the phase relationship between the inphase and quadrature components of the incoming signal. If we have a phase distorting filter in the demodulation chain, the entire detection process goes for a toss. Hence, we have to pay attention to the phase characteristics of such filters. To have no phase distortion when processing a signal through a filter, every spectral component of the signal inside the passband shoulc delayed by the same amount time delay measured in samples. In other words, the phase response $\phi(\omega)$ in the passband should be a linear function (straight line) of frequency (except for the phase wraps at the band edges). A filter that satisfies this property is called a ***linear phase filter***. FIR filters provide perfect linear phase characteristic in the passband region (Figure 5) and hence avoids phase distortion. All IIR filters provide non-linear phase characteristic. If a real-time application warrants for zero phase distortion, FIR filters are the immediate choice for design.

It is intuitive to see the phase response of a generalized linear phase filter should follow the relationship $\phi(\omega) = -m\omega + c$, where m is the slope and c is the intercept when viewing the linear relationship between the frequency and the phase response in the passband (Figure 5). The ***phase delay*** and ***group delay*** are the important filter characteristics considered for ascertaining the phase distortion and they relate to the intercept c and the slope m of the phase response in the passband. Linear phase filters are characterized by constant group delay. Any deviation in the group delay from the constant value inside the passband, indicates presence of certain degree of non-linearity in the phase and hence causes phase distortion.

Phase delay is the time delay experienced by each spectral component of the input signal. For a filter with the frequency response $H(\omega)$, the phase delay response τ_p defined in terms of phase response $\phi(\omega) = \angle H(\omega)$ as

$$\tau_p(\omega) = -\frac{\phi(\omega)}{\omega} \qquad (5)$$

Group delay is the delay experienced by a group of spectral components within a narrow frequency interval about ω [4]. The group delay response $\tau_g(\omega)$ is defined as the negative derivative of the phase response ω.

$$\tau_g(\omega) = -\frac{d}{d\omega}\phi(\omega) \qquad (6)$$

For the generalized linear phase filter, the phase delay and group delay are given by

$$\tau_g(\omega) = -\frac{d}{d\omega}\phi(\omega) = m \qquad (7)$$

$$\tau_p(\omega) = -\frac{\phi(\omega)}{\omega} = m - \frac{c}{\omega} \qquad (8)$$

Summary of design choices

• IIR filters are efficient, they can provide similar magnitude response for fewer coefficients or lower sidelobes for same number of coefficients
• For linear phase requirement, FIR filters are the immediate choice for the design
• FIR filters are inherently stable. IIR filters are susceptible to finite length words effects of fixed point arithmetic and hence the design has to be rigorously tested for stability.
• IIR filters provide less average delay compared to its equivalent FIR counterpart. If the filter has to be used in a feedback path in a system, the amount of filter delay is very critical as it affects the stability of the overall system.
• Given a specification, an IIR design can be easily deduced based on closed-form expressions. However, satisfying the design requirements using an FIR design generally requires iterative procedures.

Rate this article: ★★★★☆ (**10** votes, average: **4.60** out of 5)

References:

[1] J. H. McClellan, T. W. Parks, and L. R. Rabiner, "A Computer Program for Designing Optimum FIR Linear Phase Digital Filters," IEEE Trans, on Audio and Electroacoustics, Vol. AU-21, No. 6, pp. 506-526, December 1973.↗

[2] Frank R. Kschischang, 'The Hilbert Transform', Department of Electrical and Computer Engineering, University of Toronto, October 22, 2006.↗

[3] Thierry Turletti, 'GMSK in a nutshell', Telemedia Networks and Systems Group Laboratory for Computer Science, Massachussets Institute of Technology April, 96.↗

[4] ~~Julius O. Smith III, 'Introduction to digital filters – with audio applications', Center for Computer Research in Music and Acoustics (CCRMA), Department of Music, Stanford University.↗~~

Topics in this chapter

Books by the author

Search this site

Search ...

Study from Home Promotion

30% **discount** is given when ***all the three ebooks are checked* out in a single purchase (offer valid for a limited period).**

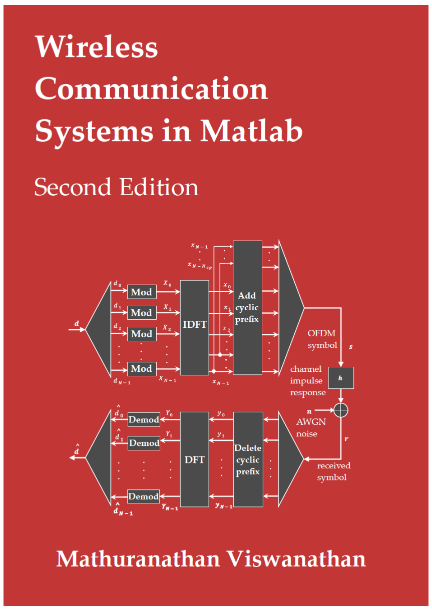
To avail the discount – use coupon code "**BESAFE**"(without quotes) when checking out all three ebooks.

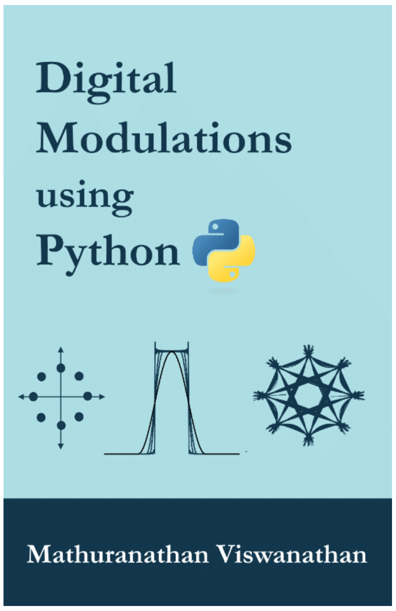
Discount not applicable for individual purchase of ebooks. Discount can only be availed during checkout.

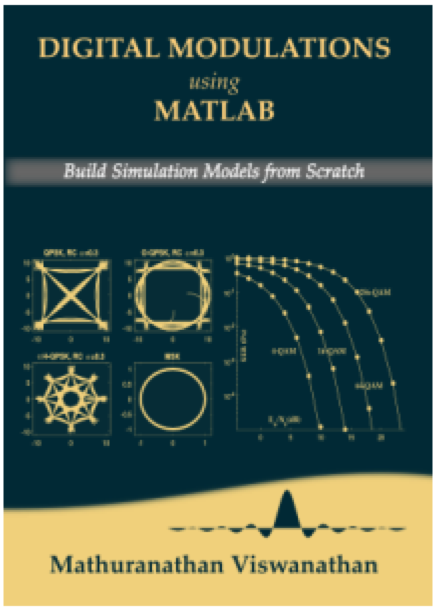
Digital Modulations using Python
Mathuranathan Viswanathan
Wireless Communication Systems in Matlab
Second Edition
Mathuranathan Viswanathan
DIGITAL MODULATIONS using MATLAB
Build Simulation Models from Scratch
Mathuranathan Viswanathan

Categories

8-PSK (1)


Wireless Communication Systems in Matlab
Second Edition
Mathuranathan Viswanathan




DIGITAL MODULATIONS using MATLAB
Build Simulation Models from Scratch
Mathuranathan Viswanathan

Wireless Communication Systems in Matlab Second Edition(PDF)
★★★★☆ (98 votes, average: 4.06 out of 5)

$14.99 – Add to Cart

Digital Modulations using Python (PDF ebook)
★★★★☆ (24 votes, average: 4.08 out of 5)

$14.99 – Add to Cart

Digital Modulations using Matlab (PDF ebook)
★★★★☆ (79 votes, average: 4.19 out of 5)

$14.99 – Add to Cart

Hand-picked Best books on Communication Engineering
Best books on Signal Processing

Latest Articles, Signal Processing, Tips & Tricks

filter design, FIR filter, IIR filter, Signal Processing

‹ Foundations of Signal Processing – Martin Vetterli, Jelena Kovacevic, and Vivek K Goyal

› Understanding Analytic Signal and Hilbert Transform

Post your valuable comments !!!

Enter your comment here...

Audio signal processing (3)
Book reviews (3)
BPSK (6)
Channel Coding (16)
Channel Modelling (38)
Constellations (4)
Correlative Coding (6)
Digital Modulations (23)
Digital Modulations using Matlab (7)
DPSK (1)
Estimation Theory (35)
Free Books (1)
GMSK (2)
Hamming Codes (1)
Inter Symbol Interference (14)
Interleaver (3)
Latest Articles (191)
Line Coding (1)
M-PSK (2)
M-QAM (4)
Machine learning (5)
Matlab Codes (74)
MIMO systems (9)
Minimum Shift keying (MSK) (1)
Nyquist (13)
OFDM (7)
Phased Array Antenna (4)
Probability (20)
Pulse Shaping (18)
Python (26)
QPSK (4)
Random Process (25)
Reed Solomon codes (4)
Shannon Theorem (4)
Signal Processing (66)

Software Defined Radio (2)
Source Coding (6)
Spread Spectrum (5)
Tips & Tricks (33)
Tutorials (16)
Uncategorized (1)
User submitted Codes (2)
VLSI (1)

Interesting blogs

- Cyclostationary Blog
- Complex to Real
- DSP Related
- DSP guide

© 2020 GaussianWaves • Powered by GeneratePress