



# Functions — By Category

<a href="#">Signal Sources</a>	Sources of random signals
<a href="#">Performance Evaluation</a>	Analyzing and visualizing performance of a communication system
<a href="#">Source Coding</a>	Quantization, companders, and other kinds of source coding
<a href="#">Error-Control Coding</a>	Block and convolutional coding
<a href="#">Interleaving/Deinterleaving</a>	Block and convolutional interleaving
<a href="#">Analog Modulation/Demodulation</a>	Passband amplitude, frequency, and phase modulation
<a href="#">Digital Modulation/Demodulation</a>	Baseband digital modulation
<a href="#">Pulse Shaping</a>	Oversampling and shaping a signal
<a href="#">Special Filters</a>	Raised cosine and Hilbert filters
<a href="#">Channels</a>	Channel models for real, complex, and binary signals
<a href="#">Equalizers</a>	Adaptive and MLSE equalizers
<a href="#">Galois Field Computations</a>	Manipulating elements of finite fields of even order
<a href="#">Computations in Galois Fields of Odd Characteristic</a>	Manipulating elements of finite fields of odd order
<a href="#">Utilities</a>	Miscellaneous relevant functions
<a href="#">Graphical User Interface</a>	Bit error rate analysis tool

## ▲ Signal Sources

<a href="#">randerr</a>	Generate bit error patterns
<a href="#">randint</a>	Generate matrix of uniformly distributed random integers
<a href="#">randsrc</a>	Generate random matrix using prescribed alphabet
<a href="#">wgn</a>	Generate white Gaussian noise

## ▲ Performance Evaluation

<a href="#">berawgn</a>	Bit error rate (BER) for uncoded AWGN channels
<a href="#">bercoding</a>	Bit error rate (BER) for coded AWGN channels
<a href="#">berconfint</a>	Bit error rate (BER) and confidence interval of Monte Carlo simulation
<a href="#">berfading</a>	Bit error rate (BER) for Rayleigh and Rician fading channels
<a href="#">berfit</a>	Fit curve to nonsmooth empirical bit error rate (BER) data
<a href="#">bersync</a>	Bit error rate (BER) for imperfect synchronization
<a href="#">biterr</a>	Compute number of bit errors and bit error rate (BER)
<a href="#">distspec</a>	Compute distance spectrum of convolutional code
<a href="#">eyediagram</a>	Generate eye diagram
<a href="#">noisebw</a>	Equivalent noise bandwidth of filter
<a href="#">scatterplot</a>	Generate scatter plot

[semianalytic](#) Calculate bit error rate (BER) using semianalytic technique  
[symerr](#) Compute number of symbol errors and symbol error rate



## Source Coding

[arithdeco](#) Decode binary code using arithmetic decoding  
[arithenco](#) Encode sequence of symbols using arithmetic coding  
[compand](#) Source code mu-law or A-law compressor or expander  
[dpcmdeco](#) Decode using differential pulse code modulation  
[dpcmenco](#) Encode using differential pulse code modulation  
[dpcmopt](#) Optimize differential pulse code modulation parameters  
[huffmandeco](#) Huffman decoder  
[huffmandict](#) Generate Huffman code dictionary for source with known probability model  
[huffmanenco](#) Huffman encoder  
[lloyds](#) Optimize quantization parameters using Lloyd algorithm  
[quantiz](#) Produce quantization index and quantized output value



## Error-Control Coding

[bchdec](#) BCH decoder  
[bchenc](#) BCH encoder  
[bchgenpoly](#) Generator polynomial of BCH code  
[bchnumerr](#) Number of correctable errors for BCH code  
[convenc](#) Convolutionally encode binary data  
[cyclgen](#) Produce parity-check and generator matrices for cyclic code  
[cyclpoly](#) Produce generator polynomials for cyclic code  
[decode](#) Block decoder  
[dvbs2ldpc](#) Low-density parity-check codes from DVB-S.2 standard  
[encode](#) Block encoder  
[fec.ldpcdec](#) Construct LDPC decoder object  
[fec.ldpcenc](#) Construct LDPC encoder object  
[gen2par](#) Convert between parity-check and generator matrices  
[gfweight](#) Calculate minimum distance of linear block code  
[hammgen](#) Produce parity-check and generator matrices for Hamming code  
[rsdec](#) Reed-Solomon decoder  
[rsdecof](#) Decode ASCII file encoded using Reed-Solomon code  
[rsenc](#) Reed-Solomon encoder  
[rsencof](#) Encode ASCII file using Reed-Solomon code  
[rsgenpoly](#) Generator polynomial of Reed-Solomon code  
[syndtable](#) Produce syndrome decoding table  
[vitdec](#) Convolutionally decode binary data using Viterbi algorithm



## Interleaving/Deinterleaving

[algdeintrlv](#) Restore ordering of symbols using algebraically derived permutation table  
[algintrlv](#) Reorder symbols using algebraically derived permutation table  
[convdeintrlv](#) Restore ordering of symbols using shift registers  
[convintrlv](#) Permute symbols using shift registers  
[deintrlv](#) Restore ordering of symbols

<a href="#">heldeintrlv</a>	Restore ordering of symbols permuted using <code>helintrlv</code>
<a href="#">helintrlv</a>	Permute symbols using helical array
<a href="#">helscandintrlv</a>	Restore ordering of symbols in helical pattern
<a href="#">helscanintrlv</a>	Reorder symbols in helical pattern
<a href="#">intrlv</a>	Reorder sequence of symbols
<a href="#">matdeintrlv</a>	Restore ordering of symbols by filling matrix by columns and emptying it by rows
<a href="#">matintrlv</a>	Reorder symbols by filling matrix by rows and emptying it by columns
<a href="#">muxdeintrlv</a>	Restore ordering of symbols using specified shift registers
<a href="#">muxintrlv</a>	Permute symbols using shift registers with specified delays
<a href="#">randdeintrlv</a>	Restore ordering of symbols using random permutation
<a href="#">randintrlv</a>	Reorder symbols using random permutation



## Analog Modulation/Demodulation

<a href="#">amdemod</a>	Amplitude demodulation
<a href="#">ammod</a>	Amplitude modulation
<a href="#">fmdemod</a>	Frequency demodulation
<a href="#">fmmod</a>	Frequency modulation
<a href="#">pmdemod</a>	Phase demodulation
<a href="#">pmmod</a>	Phase modulation
<a href="#">ssbdemod</a>	Single sideband amplitude demodulation
<a href="#">ssbmod</a>	Single sideband amplitude modulation



## Digital Modulation/Demodulation

<a href="#">dpskdemod</a>	Differential phase shift keying demodulation
<a href="#">dpskmod</a>	Differential phase shift keying modulation
<a href="#">fskdemod</a>	Frequency shift keying demodulation
<a href="#">fskmod</a>	Frequency shift keying modulation
<a href="#">genqamdemod</a>	General quadrature amplitude demodulation
<a href="#">genqammod</a>	General quadrature amplitude modulation
<a href="#">modem</a>	Package of modem classes
<a href="#">modem.pskdemod</a>	Construct PSK demodulator object
<a href="#">modem.pskmod</a>	Construct PSK modulator object
<a href="#">modem.qamdemod</a>	Construct QAM demodulator object
<a href="#">modem.qammod</a>	Construct QAM modulator object
<a href="#">modnorm</a>	Scaling factor for normalizing modulation output
<a href="#">mskdemod</a>	Minimum shift keying demodulation
<a href="#">mskmod</a>	Minimum shift keying modulation
<a href="#">oqpskdemod</a>	Offset quadrature phase shift keying demodulation
<a href="#">oqpskmod</a>	Offset quadrature phase shift keying modulation
<a href="#">pamdemod</a>	Pulse amplitude demodulation
<a href="#">pammod</a>	Pulse amplitude modulation
<a href="#">pskdemod</a>	Phase shift keying demodulation
<a href="#">pskmod</a>	Phase shift keying modulation
<a href="#">qamdemod</a>	Quadrature amplitude demodulation
<a href="#">qammod</a>	Quadrature amplitude modulation



## Pulse Shaping

<a href="#">intdump</a>	Integrate and dump
<a href="#">rcosflt</a>	Filter input signal using raised cosine filter
<a href="#">rectpulse</a>	Rectangular pulse shaping



## Special Filters

<a href="#">hank2sys</a>	Convert Hankel matrix to linear system model
<a href="#">hilbiir</a>	Design Hilbert transform IIR filter
<a href="#">rcosine</a>	Design raised cosine filter



### Lower Level Functions for Special Filters

<a href="#">rcosfir</a>	Design raised cosine finite impulse response (FIR) filter
<a href="#">rcosiir</a>	Design raised cosine infinite impulse response (IIR) filter



## Channels

<a href="#">awgn</a>	Add white Gaussian noise to signal
<a href="#">bsc</a>	Model binary symmetric channel
<a href="#">doppler</a>	Package of Doppler classes
<a href="#">doppler.flat</a>	Construct flat Doppler spectrum object
<a href="#">doppler.gaussian</a>	Construct Gaussian Doppler spectrum object
<a href="#">doppler.jakes</a>	Construct Jakes Doppler spectrum object
<a href="#">doppler.rjakes</a>	Construct restricted Jakes Doppler spectrum object
<a href="#">doppler.rounded</a>	Construct rounded Doppler spectrum object
<a href="#">filter (channel)</a>	Filter signal with channel object
<a href="#">plot (channel)</a>	Plot channel characteristics with channel visualization tool
<a href="#">rayleighchan</a>	Construct Rayleigh fading channel object
<a href="#">reset (channel)</a>	Reset channel object
<a href="#">ricianchan</a>	Construct Rician fading channel object



## Equalizers

<a href="#">cma</a>	Construct constant modulus algorithm (CMA) object
<a href="#">dfe</a>	Construct decision-feedback equalizer object
<a href="#">equalize</a>	Equalize signal using equalizer object
<a href="#">lineareq</a>	Construct linear equalizer object
<a href="#">lms</a>	Construct least mean square (LMS) adaptive algorithm object
<a href="#">mlseeq</a>	Equalize linearly modulated signal using Viterbi algorithm
<a href="#">normlms</a>	Construct normalized least mean square (LMS) adaptive algorithm object
<a href="#">reset (equalizer)</a>	Reset equalizer object
<a href="#">rls</a>	Construct recursive least squares (RLS) adaptive algorithm object
<a href="#">signlms</a>	Construct signed least mean square (LMS) adaptive algorithm object
<a href="#">varlms</a>	Construct variable-step-size least mean square (LMS) adaptive algorithm object



## Galois Field Computations

<a href="#">convmtx</a>	Convolution matrix of Galois field vector
<a href="#">cosets</a>	Produce cyclotomic cosets for Galois field
<a href="#">dftmtx</a>	Discrete Fourier transform matrix in Galois field
<a href="#">fft</a>	Discrete Fourier transform
<a href="#">filter (gf)</a>	1-D digital filter over Galois field
<a href="#">gf</a>	Create Galois field array
<a href="#">gftable</a>	Generate file to accelerate Galois field computations
<a href="#">ifft</a>	Inverse discrete Fourier transform
<a href="#">isprimitive</a>	True for primitive polynomial for Galois field
<a href="#">log</a>	Logarithm in Galois field
<a href="#">minpol</a>	Find minimal polynomial of Galois field element
<a href="#">mldivide</a>	Matrix left division \ of Galois arrays
<a href="#">primpoly</a>	Find primitive polynomials for Galois field



### Additional MATLAB Functions and Operators

+ -	Addition and subtraction of Galois arrays
* / \	Matrix multiplication and division of Galois arrays
. * ./ . \	Elementwise multiplication and division of Galois arrays
^	Matrix exponentiation of Galois array
. ^	Elementwise exponentiation of Galois array
' .'	Transpose of Galois array
==, ~=	Relational operators for Galois arrays
<a href="#">all</a>	True if all elements of a Galois vector are nonzero
<a href="#">any</a>	True if any element of a Galois vector is nonzero
<a href="#">conv</a>	Convolution of Galois vectors
<a href="#">deconv</a>	Deconvolution and polynomial division
<a href="#">det</a>	Determinant of square Galois matrix
<a href="#">diag</a>	Diagonal Galois matrices and diagonals of a Galois matrix
<a href="#">inv</a>	Inverse of Galois matrix
<a href="#">isempty</a>	True for empty Galois arrays
<a href="#">length</a>	Length of Galois vector
<a href="#">lu</a>	Lower-upper triangular factorization of Galois array
<a href="#">polyval</a>	Evaluate polynomial in Galois field
<a href="#">rank</a>	Rank of a Galois array
<a href="#">reshape</a>	Reshape Galois array
<a href="#">roots</a>	Find polynomial roots across a Galois field
<a href="#">size</a>	Size of Galois array
<a href="#">tril</a>	Extract lower triangular part of Galois array
<a href="#">triu</a>	Extract upper triangular part of Galois array



### Computations in Galois Fields of Odd Characteristic

<a href="#">gfadd</a>	Add polynomials over Galois field
<a href="#">gfconv</a>	Multiply polynomials over Galois field
<a href="#">gfcosets</a>	Produce cyclotomic cosets for Galois field
<a href="#">gfdeconv</a>	Divide polynomials over Galois field
<a href="#">gfdiv</a>	Divide elements of Galois field
<a href="#">gffilter</a>	Filter data using polynomials over prime Galois field
<a href="#">gflineq</a>	Find particular solution of $Ax = b$ over prime Galois field
<a href="#">gfminpol</a>	Find minimal polynomial of Galois field element
<a href="#">gfmul</a>	Multiply elements of Galois field
<a href="#">gfpretty</a>	Polynomial in traditional format
<a href="#">gfprimck</a>	Check whether polynomial over Galois field is primitive
<a href="#">gfprimdf</a>	Provide default primitive polynomials for Galois field
<a href="#">gfprimfd</a>	Find primitive polynomials for Galois field
<a href="#">gfrank</a>	Compute rank of matrix over Galois field
<a href="#">gfrepcov</a>	Convert one binary polynomial representation to another
<a href="#">gfroots</a>	Find roots of polynomial over prime Galois field
<a href="#">gfsub</a>	Subtract polynomials over Galois field
<a href="#">gftrunc</a>	Minimize length of polynomial representation
<a href="#">gftuple</a>	Simplify or convert Galois field element formatting



## Utilities

<a href="#">alignsignals</a>	Aligns two signals by delaying earliest signal
<a href="#">bi2de</a>	Convert binary vectors to decimal numbers
<a href="#">bin2gray</a>	Convert positive integers into corresponding Gray-encoded integers
<a href="#">de2bi</a>	Convert decimal numbers to binary vectors
<a href="#">finddelay</a>	Estimates delay(s) between signals
<a href="#">gray2bin</a>	Convert Gray-encoded positive integers to corresponding Gray-decoded integers
<a href="#">iscatastrophic</a>	True for trellis corresponding to catastrophic convolutional code
<a href="#">istrellis</a>	True for valid trellis structure
<a href="#">marcumq</a>	Generalized Marcum Q function
<a href="#">mask2shift</a>	Convert mask vector to shift for shift register configuration
<a href="#">oct2dec</a>	Convert octal to decimal numbers
<a href="#">poly2trellis</a>	Convert convolutional code polynomials to trellis description
<a href="#">qfunc</a>	Q function
<a href="#">qfuncinv</a>	Inverse Q function
<a href="#">seqgen</a>	Sequence generator package
<a href="#">seqgen.pn</a>	Constructs default PN sequence generator object
<a href="#">shift2mask</a>	Convert shift to mask vector for shift register configuration
<a href="#">vec2mat</a>	Convert vector into matrix



## Additional MATLAB Functions

<a href="#">erf</a>	Error function
<a href="#">erfc</a>	Complementary error function



## Graphical User Interface

[bertool](#)

Open bit error rate analysis GUI (BERTool)

 Selected Bibliography for Galois Fields

Functions — Alphabetical List 

© 1984-2007 The MathWorks, Inc. · [Terms of Use](#) · [Patents](#) · [Trademarks](#) · [Acknowledgments](#)