Digital

August 2008, 372504F-01

Use the VIs on the **Digital Modulation** palette to modulate and demodulate signals using PSK, FSK, QAM, MSK, ASK, and CPM formats, and for adding common impairments.

Palette Object	Description
<u>MT Open</u> <u>Session</u>	Creates an empty modulation session refnum. When the modulation session refnum is created, navigate to the All Functions » Application Control palette and drop an empty property node on the block diagram. When you wire the modulation session refnum into the property node, the property node displays the name Modulation Toolkit as its title. You can either write (set) or read (get) attributes using the property node.
<u>MT</u> <u>Close</u> <u>Session</u>	Closes any open modulation session.
<u>MT</u> <u>Generate</u> <u>Bits</u> (poly)	Generates the sequence of data bits to be <u>modulated</u> . This polymorphic VI can generate Fibonacci or Galois pseudonoise (PN) bit sequences. It can also generate bit sequences based on a user-defined pattern. The selected pattern is repeated until the user-specified number of total bits is generated.

Subpalette	Description
<u>Coding</u>	Use the VIs on the Coding palette to encode, spread, and puncture baseband data streams.
Interleaving	Use the VIs on the Interleaving palette to perform permutation, and block and convolutional interleaving.
<u>Modulation</u>	Use the VIs on the Modulation palette to implement several types of digital modulation.
Decoding	Use the VIs on the Decoding palette to decode, despread, and depuncture baseband data streams.
DeInterleaving	Use the VIs on the Deinterleaving palette to perform

	depermutation, and block and convolutional deinterleaving.
<u>Demodulation</u>	Use the VIs on the Demodulation palette to implement several types of digital demodulation.
Impairments	Use the VIs on the Impairments palette to add several types of baseband impairments to digital signals.
<u>Equalization</u>	Use the VIs on the Equalization palette to implement several types of digital equalization.
<u>Utilities</u>	Use the VIs on the Utilities palette to perform tasks such as applying filters, generating synchronization parameters and symbol maps, and opening and closing your session with the Modulation Toolkit VIs.
<u>Measurements</u>	Use the VIs on the Measurements palette to measure parameters including BER, Rho, burst timing, and quadrature impairments.
<u>Visualization</u>	Use the VIs on the Visualization palette to create eye and constellation diagrams, IQ-to-XY graphs, and trellis diagrams.

 $\ensuremath{\textcircled{\sc 0}}$ 2008 National Instruments Corporation. All Rights Reserved.

Signal Continuability

In some cases, Modulation Toolkit palette-level VIs are continuable, meaning that they can operate on continuous signals that have been broken into blocks or frames. This division can separate the large dataset of the digitized signal into smaller blocks. Breaking signals into blocks or frames can help in situations where there are PC memory limitations or a limited amount of hardware acquisition memory.

Because Modulation Toolkit VIs are continuable, you can utilize them in two distinct modes:

- **One-shot mode**—Processes all data with a single call to the appropriate Modulation Toolkit VIs.
- **Continuous mode**—Processes data block-by-block as available or appropriate.

When using continuous mode, consider the action of VIs that include a **reset?** parameter. This parameter is present in VIs that store state information when called in a loop, as when processing a continuous stream of data. In these situations, the VI is typically reset when initially called, and on subsequent calls information is retained about data processed in prior calls to the same instance of the VI. After several iterations, you can reinitialize the VI to return it to a known state. The MT <u>Generate Bits (poly)</u> VI, shown in the following figure, is an example of a VI that utilizes the **reset?** parameter in this way. The VI can generate a continuous PN sequence, which is a pseudorandom bit stream that is seeded for easy continuous reproduction.



MT Generate Bits VI in One-Shot Mode



MT Generate Bits VI in Continuous Mode

The preceding examples show two ways you can call the <u>MT Generate</u> <u>Bits (poly)</u> VI. The top section shows the VI being called a single time to generate the first 32 bits of a PN sequence. The bottom section shows the VI being called inside a For loop and generating the same 32 bits by concatenating two 16-bit chunks. Because the MT Generate Bits VI is continuable, the same instance can be called multiple times, with the first call being reset and subsequent calls maintaining state information from previous calls so as to continue each new operation where the last call left off.

- Note By default, all continuable Modulation Toolkit VIs reset when an instance of the VI is first called and operate thereafter in a continuable manner. The **reset?** parameter default values are set accordingly, which means that the VIs in the preceding example would reset automatically even if the **reset?** parameters were not explicitly wired.
- **Note** Some Modulation Toolkit VIs require a minimum input block length before they can run. After a reset condition, these VIs do not return data until the minimum input length has been exceeded. For instance, the <u>MT Convolutional Decoder</u> VI requires at least $D \times n$ input bits (where *D* is the traceback depth and *n* is the code word length) after a reset before it can begin to generate decoded bits. Each call to the VI with less than the appropriate input length results in the input data being buffered and appended to any input data from previous iterations that followed the last reset.

MT Generate Bits (poly) VI

Generates the sequence of data bits to be <u>modulated</u>. This polymorphic VI can generate Fibonacci or Galois pseudonoise (PN) bit sequences. It can also generate bit sequences based on a user-defined pattern. The selected pattern is repeated until the user-specified number of **total bits** is generated.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

<u>Details</u>

MT Generate Bits (User Defined)

This polymorphic instance generates bit sequences based on a userdefined pattern. The selected pattern is repeated until the user-specified number of total bits is generated.



- **total bits** specifies the total number of pseudorandom bits to be generated. The default is 128.
- user base bit pattern specifies a user-defined base pattern of bits. The default is empty.
- **reset?** specifies whether to continue generating bits using the previous iteration states. **reset?** must be set to TRUE when initializing the PN generator with a new PN seed. When **reset?** is set to FALSE, the PN sequence generator resumes where it stopped during the previous iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - source describes the origin of the error or warning. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

- **(III)** output bit stream returns the user-defined binary data stream corresponding to the user base bit pattern.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate Bits (Fibonacci, Primitive Polynomial)

This polymorphic instance generates Fibonacci pseudonoise (PN) bit sequences. The selected pattern is repeated until the user-specified number of total bits is generated.



- **total bits** specifies the total number of pseudorandom bits to be generated. The default is 128.
- **Specify primitive polynomial** specifies the primitive polynomial for the PN bit sequence to be generated. The degree of the primitive polynomial determines the PN order. The primitive polynomial is specified by an 8-bit signed integer array. If the degree of the primitive polynomial is *N*, for example, $p(x) = a0+a1x+a2x^2 + \dots + a$ *N* a ^{*N*}, the array contains (*N*+1) elements. The first element is *a*0, and the last element is *aN*. Ensure that the polynomial you provide is a primitive polynomial. For example, if the primitive polynomial is $p(x)=1+x^{14}+x^{15}$, then *N*=15, and the array contains *N*+1=16 elements.
- **seed in** specifies the initial state of the PN generator shift register. If no seed is specified, the default seed is used. The default is 0xD6BF7DF2.
- **reset?** specifies whether to continue generating bits using the previous iteration states. **reset?** must be set to TRUE when initializing the PN generator with a new PN seed. When **reset?** is set to FALSE, the PN sequence generator resumes where it stopped during the previous iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **i i output bit stream** returns the generated pseudorandom data bits. If the PN sequence order is *N*, the output data is periodic with period $T = 2^N - 1$. For example, if N = 7, the output sequence repeats after every T = 127 bits.
- **seed out** returns a seed for use as **seed in** during the next call to this VI when **reset?** is set to FALSE.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate Bits (Fibonacci, PN Order)

This polymorphic instance generates Fibonacci pseudonoise (PN) bit sequences. The selected pattern is repeated until the user-specified number of total bits is generated.



- **total bits** specifies the total number of pseudorandom bits to be generated. The default is 128.
- **PN sequence order** specifies the order of the PN bit sequence to be generated. Valid values are 5–31, inclusive. The default is 9.
- **seed in** specifies the initial state of the PN generator shift register. If no seed is specified, the default seed is used. The default is 0xD6BF7DF2.
- **reset?** specifies whether to continue generating bits using the previous iteration states. **reset?** must be set to TRUE when initializing the PN generator with a new PN seed. When **reset?** is set to FALSE, the PN sequence generator resumes where it stopped during the previous iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(iii)** output bit stream returns the generated pseudorandom data bits. If the **PN sequence order** is *N*, the output data is periodic with period $T = 2^{N}-1$. For example, if N = 7, the output sequence repeats after every T = 127 bits.
- **seed out** returns a seed for use as **seed in** during the next call to this VI when **reset?** is set to FALSE.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate Bits (Galois, Primitive Polynomial)

This polymorphic instance generates Galois pseudonoise (PN) bit sequences. The selected pattern is repeated until the user-specified number of total bits is generated.



- **total bits** specifies the total number of pseudorandom bits to be generated. The default is 128.
- **Specify primitive polynomial** specifies the primitive polynomial for the PN bit sequence to be generated. The degree of the primitive polynomial determines the PN order. The primitive polynomial is specified by an 8-bit signed integer array. If the degree of the primitive polynomial is *N*, for example, $p(x) = a0+a1x+a2x^2 + \dots + a$ *N* a ^{*N*}, the array contains (*N*+1) elements. The first element is *a*0, and the last element is *aN*. Ensure that the polynomial you provide is a primitive polynomial. For example, if the primitive polynomial is $p(x)=1+x^{14}+x^{15}$, then *N*=15, and the array contains *N*+1=16 elements.
- **seed in** specifies the initial state of the PN generator shift register. If no seed is specified, the default seed is used. The default is 0xD6BF7DF2.
- **reset?** specifies whether to continue generating bits using the previous iteration states. **reset?** must be set to TRUE when initializing the PN generator with a new PN seed. When **reset?** is set to FALSE, the PN sequence generator resumes where it stopped during the previous iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **i i output bit stream** returns the generated pseudorandom data bits. If the PN sequence order is *N*, the output data is periodic with period $T = 2^N - 1$. For example, if N = 7, the output sequence repeats after every T = 127 bits.
- **seed out** returns a seed for use as **seed in** during the next call to this VI when **reset?** is set to FALSE.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate Bits (Galois, PN Order)

This polymorphic instance generates Galois pseudonoise (PN) bit sequences. The selected pattern is repeated until the user-specified number of total bits is generated.



- **total bits** specifies the total number of pseudorandom bits to be generated. The default is 128.
- **PN sequence order** specifies the order of the PN bit sequence to be generated. Valid values are 5–31, inclusive.
- **seed in** specifies the initial state of the PN generator shift register. If no seed is specified, the default seed is used. The default is 0xD6BF7DF2.
- **reset?** specifies whether to continue generating bits using the previous iteration states. **reset?** must be set to TRUE when initializing the PN generator with a new PN seed. When **reset?** is set to FALSE, the PN sequence generator resumes where it stopped during the previous iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(iii)** output bit stream returns the generated pseudorandom data bits. If the **PN sequence order** is *N*, the output data is periodic with period $T = 2^{N}-1$. For example, if N = 7, the output sequence repeats after every T = 127 bits.
- **seed out** returns a seed for use as **seed in** during the next call to this VI when **reset?** is set to FALSE.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Pseudorandom or pseudonoise (PN) sequences, though deterministic in nature, satisfy many properties (autocorrelation, crosscorrelation, and so on) of random numbers. PN sequences are used in many applications and standards such as 802.11a and DVB. Some examples of PN sequences are *m*-sequences (also called maximal length shift register sequences), Gold sequences, and Kasami sequences. An *m*-sequence generates a periodic sequence of length $L = 2^m - 1$ bits and is generated by Linear Feedback Shift Registers (LFSR). Two very well known implementations of *m*-sequences are the Fibonacci implementation and the Galois implementation.





Galois Implementation

The preceding figures shows the Fibonacci and Galois implementations of maximal length shift register *m*-sequences. As can be seen in these figures, *m*-sequences contain *m* shift registers. The shift register set is filled with an *m*-bit initial seed that can be any value except 0 (if the *m* bits in the *m* shift registers are all zero, then it is a degenerate case and the output of the generator is 0).

The following examples demonstrate bit generation.

1. Inputs are specified as follows:

Primitive polynomial: $1 + X^{14} + X^{15}$ Initial seed: 00000010101001 The circuitry is shown in the following figure.



The preceding example depicts the Fibonacci implementation. This structure is used in different standards, including DVB.

2. Inputs are specified as follows:

Primitive polynomial: $1 + X^{14} + X^{15}$

Initial seed: 00000010101001

The circuitry is shown in the following figure:



Seed	Output
00000010101001	0
000000101010010	0
000001010100100	0
000010101001000	0
000101010010000	0

001010100100000	0
010101001000000	0
10101001000000	1
11010010000001	1
00100100000011	0
01001000000110	0
10010000001100	1
10100000011001	1
11000000110011	1
00000001100111	0
00000011001110	0

The preceding example illustrates how PN bits are generated using the Galois implementation.

References

- 1. McEliece, Robert J. Finite Fields for Computer Scientists and Engineers (The Kluwer International Series in Engineering and Computer Science). New York: Springer Publishers, 1986.
- 2. Press, William H., et al. *Numerical Recipes in C: The Art of Scientific Computing*. 2nd ed. Cambridge: Cambridge University Press, 1992.
- 3. Proakis, John G. and Masoud Salehi. *Communication Systems Engineering.* 2d ed. Englewood Cliffs: Prentice-Hall, 2001.
- 4. Sklar, Bernard. *Digital Communications: Fundamentals and Applications.* 2d ed. Englewood Cliffs: Prentice-Hall, 2001.

Coding Owning Palette: <u>Digital</u>

Use the VIs on the **Coding** palette to encode, spread, and puncture baseband data streams.

Palette Object	Description
<u>MT Spread</u> <u>Symbols</u>	Performs the <u>DSSS</u> spreading operation given a user- defined spreading code. The spreading code algorithm performs non-return-to-zero (NRZ) encoding of the input bit stream and the spreading code . However, the output is an array of zeros and ones.
MT Convolutional Encode	Generates an encoded bit stream based on a specified code rate. The code rate is equal to the ratio of the data word length to the code word length. This VI allows you to choose a code rate of 1/2, 1/3, 1/4, 2/3, or 3/4 with preconfigured generator matrices. Use the Generator Matrix instance to specify custom values.
<u>MT Reed</u> <u>Solomon</u> <u>Symbol</u> <u>Encode</u>	Generates a normal or shortened Reed-Solomon (RS) encoded integer symbol stream.
<u>MT Golay</u> <u>Encoder</u>	Generates a Golay-encoded bit stream. The VI provides the two triple-correcting Golay codes: the Golay (23,12,3) code and the extended Golay (24,12,3) code.
MT Hamming Encoder	Generates a Hamming-encoded bit stream.
MT BCH Encoder	Generates a Bose-Chaudhari-Hocquenghem (BCH)- encoded bit stream. The BCH encoding algorithm produces the BCH code word using a systematic encoding scheme. User-specified values of <i>n</i> , <i>k</i> , and <i>t</i> define the code word length, data word length and error correcting capacity, respectively, in bits. You can choose between well-known (n, k, t) combinations or specify custom (n, k, t) values. The encoding scheme takes in a bit data word of <i>k</i> bits and produces a code word of <i>n</i> bits.

MT PuncturePunctures the input data stream and returns an outputData Streamdata stream that is at a higher rate relative to the input.

Subpalette	Description
<u>LDPC</u>	Use the LDPC subpalette to generate regular and irregular parity check matrices and carry out LDPC encoding on a bitstream.

LDPC

Owning Palette: Coding

Use the **LDPC** subpalette to generate regular and irregular parity check matrices and carry out LDPC encoding on a bitstream.

Palette Object	Description
MT LDPC Generate Irregular Parity Check Matrix	Generates an irregular parity check matrix that is used by the <u>MT LDPC Encoder</u> VI for <u>LDPC encoding</u> . In this VI, the number of ones in each column is kept constant, and the number of ones in each row is random.
MT LDPC Generate Regular Parity Check Matrix	Generates an irregular parity check matrix that is used by the <u>MT LDPC Encoder</u> VI for <u>LDPC encoding</u> . In this VI, the number of ones in each column and number of ones in each row are kept constant.
<u>MT LDPC</u> <u>Encoder</u>	Performs LDPC encoding on the input bit stream based on the parity check matrix . The parity check matrix could be generated by the <u>MT LDPC Generate Regular Parity Check</u> <u>Matrix</u> VI, the <u>MT LDPC Generate Irregular Parity Check</u> <u>Matrix</u> VI, or provided as an input to this VI.

MT LDPC Encoder VI

Performs LDPC encoding on the **input bit stream** based on the **parity check matrix**. The parity check matrix could be generated by the <u>MT</u> LDPC Generate Regular Parity Check Matrix VI, the <u>MT LDPC Generate</u> Irregular Parity Check Matrix VI, or provided as an input to this VI.



parity check matrix in specifies the sparse parity check matrix. This matrix can be generated by the <u>MT LDPC Generate Regular</u> <u>Parity Check Matrix</u> VI, the <u>MT LDPC Generate Irregular Parity</u> <u>Check Matrix</u> VI, or you can specify a parity check matrix that is not rank deficient.

- **input bit stream** specifies the incoming bit stream to be mapped to LDPC symbols.
- reset? (T) specifies whether to clear the internal state of the encoder. By default, this VI resets on first call.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - source describes the origin of the error or warning. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

- **parity check matrix out** specifies the preprocessed parity check matrix that is used by the encoder for encoding. This parity check matrix should be provided to the decoder.
- **[18]** output bit stream returns the encoded data bits.
 - Note The encoder places the generated parity bits at the end of the output bit stream by default. Hence the encoded word format is [message bits | parity bits].
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT LDPC Generate Irregular Parity Check Matrix VI

Generates an irregular parity check matrix that is used by the <u>MT LDPC</u> <u>Encoder</u> VI for <u>LDPC encoding</u>. In this VI, the number of ones in each column is kept constant, and the number of ones in each row is random.



- **n** specifies the number of columns.
- **m** specifies the number of rows.
- **j** specifies the number of ones in the column of LDPC code. **j** must be an odd number.
- **maximum number of iterations** specifies the maximum number of iterations to return a parity check matrix that is not rank deficient.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **irregular parity check matrix** returns the irregular parity check matrix generated by this VI.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT LDPC Generate Regular Parity Check Matrix VI

Generates an irregular parity check matrix that is used by the <u>MT LDPC</u> <u>Encoder</u> VI for <u>LDPC encoding</u>. In this VI, the number of ones in each column and number of ones in each row are kept constant.



- **n** specifies the number of columns.
- **j** specifies the number of ones in the column of LDPC code. **j** must be an odd number.
- **\mathbf{I} k** specifies the number of ones in the row of LDPC code.
- **maximum number of iterations** specifies the maximum number of iterations to return a parity check matrix that is not rank deficient.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **regular parity check matrix** returns the regular parity check matrix generated by this VI.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT BCH Encoder VI

Generates a Bose-Chaudhari-Hocquenghem (BCH)-encoded bit stream. The BCH encoding algorithm produces the BCH code word using a systematic encoding scheme. User-specified values of n, k, and t define the code word length, data word length and error correcting capacity, respectively, in bits. You can choose between well-known (n, k, t) combinations or specify custom (n, k, t) values. The encoding scheme takes in a bit data word of k bits and produces a code word of n bits.

Details



- **input bit stream** specifies the bit sequence representing the data bits to encode. Use the <u>MT Generate Bits (poly)</u> VI to produce this bit stream or wire a custom data bit stream to this parameter.
 - **Tip** To ensure that the **output bit stream** returned by the BCH encoder corresponds to the entire **input bit stream**, specify an **input bit stream** with an integer length that is a multiple of the data word length *k*. If the length of the **input bit stream** is not an integer multiple of *k*, the last {length(**input bit stream**) modulo *k*} bits are buffered in the encoder. These bits are added to the beginning of the incoming bit stream during the beginning of the next iteration if **reset?** is set to FALSE.
- **n,k,t** specifies the values of *n*, *k*, and *t* that define the code word length, data word length, and error-correcting capacity, respectively, in bits. The *n*, *k*, and *t* values for BCH encoding are interrelated such that specifying two values constrains the third value. Select from a set of well-known (n,k,t) triples that are widely used for BCH encoding or specify an arbitrary triple by entering three commaseparated integers. The default is 7, 4, 1.
- **reset?** specifies whether to clear the internal state of the encoder. By default, the VI resets on first call. When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the BCH **n,k,t** values. The VI also clears any buffered bits from previous

iterations. When **reset?** is set to FALSE, the VI continues performing BCH encoding from the previous iteration. Any buffered bits from the previous iteration are added to the beginning of the **input bit stream** prior to performing BCH encoding. The default is TRUE.

- **Tip** When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the user-specified **n,k,t** combination. When the Galois field structure is initialized, this same field structure is used for performing the BCH encoding provided **reset?** is set to FALSE. This initialization may consume significant amounts of time for values of *n* greater than 127. Therefore, to get the best performance from the BCH encoder, NI recommends you perform BCH encoding with **reset?** set to TRUE for the first iteration and set to FALSE thereafter.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the decoded bit stream.

error out passes error or warning information out of a VI to be used

by other VIs. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The prepopulated **n,k,t** parameter contains all valid codes for values of *n* up to 511. However, the Modulation Toolkit supports BCH encoding and decoding up to Galois field sizes of GF(65,536), so you can specify BCH code word lengths up to 65,535. You can specify code word lengths up to 65,535 by entering valid values of the code word length (*n*), data word length (*k*), and error correction capacity (*t*) separated by commas inside the string combo box.



Note

Not all **n,k,t** combinations specify valid BCH codes. The BCH encoder/decoder VIs perform validation for values of the code word length (*n*). If *n* is not of the form 2^m -1, the VI returns an error. The only validation criterion for the data word length (*k*) and error-correcting capacity (*t*) is that they should be positive integers strictly lesser than *n*. The Modulation Toolkit does not guarantee specifications of BCH **n,k,t** values that satisfy the above criterion but are not valid BCH codes.

Values of standard (n,k,t) triples are defined in Wicker, Stephen B. *Error Control Systems for Digital Communications and Storage*. Englewood Cliffs: Prentice-Hall, 1995.

MT Convolutional Encode VI

Generates an encoded bit stream based on a specified code rate. The code rate is equal to the ratio of the data word length to the code word length. This VI allows you to choose a code rate of 1/2, 1/3, 1/4, 2/3, or 3/4 with preconfigured generator matrices. Use the Generator Matrix instance to specify custom values.

Use the pull-down menu to select an instance of this VI.

Select an instance

<u>Details</u>

MT Convolutional Encoder (Generator Matrix)

This polymorphic instance generates an encoded bit stream based on a specified generator matrix.



- **input bit stream** specifies the bit sequence representing the data word to encode. Use the <u>MT Generate Bits (poly)</u> VI to produce this bit stream or wire a custom data bit stream to this parameter.
- **generator matrix (n x k)** specifies the generator connection polynomial matrix used for setting the convolutional feedforward encoder connections in octal format.

The convolutional encoder is modeled as a linear feedforward shift register arrangement consisting of *k* rows with *K*-1 shift registers per row, where *k* denotes the data word length and *K* denotes the constraint length. If a_{ij} { $0 \le i \le n-1$, $0 \le j \le k-1$ } denotes a particular element in the generator matrix, the row index *i* corresponds to the convolutional encoder output y_i that is affected by this element, while the column index *j* corresponds to the *j*th row in the *k* row shift register arrangement. Thus a_{ij} specifies how the *K* bits in the *j*th row of the feed forward shift register matrix affects the *i*th output of

5

the convolutional encoder. The default is \lfloor^7 .

For a rate 2/3 convolutional encoder, the generator matrix is specified as

 $\begin{bmatrix} 100100 & 011000 \\ 011100 & 101000 \\ 110000 & 010000 \end{bmatrix} = \begin{bmatrix} 44 & 30 \\ 34 & 50 \\ 60 & 20 \end{bmatrix}$

Notice that the matrix on the right represents the elements in octal format. Zeros are padded at the end of the corresponding code generator sequences such that their total length is a multiple of three digits. The following diagram depicts the rate 2/3 convolutional encoder corresponding to the previously mentioned

generator matrix, with constraint length equal to 4. In this diagram, **D** represents a shift register or memory element.



Here, y_i^{j} , $0 \le j \le n-1$ denotes the *j*th output of the convolutional encoder, in the *i*th encoding instance.

- **constraint length (K)** specifies the maximum number of encoded bits that can be affected by a single input bit. This value represents (1 + maximal memory order), where maximal memory order refers to the length of the longest shift register chain in the convolutional encoder. The default is 3.
- **initial state** specifies the encoder initialization state for the k(K-1) shift registers, where k is the input data word length and K specifies the constraint length. On the first call to this VI, and thereafter when **reset?** is set to TRUE (the default), the encoder state is reset to this value on each call to this VI. When **reset?** is set to FALSE, the final state from the previous iteration is used in subsequent calls to this VI and the **initial state** parameter is ignored. The default is 0.
- reset? specifies whether the internal state of the convolutional encoder is cleared. By default, the VI resets on first call. When reset? is set to TRUE, this VI clears any buffered bits from previous iterations, and the encoder is initialized to start from the specified initial state. When reset? is set to FALSE, this VI continues performing convolutional encoding from the previous iteration. Any buffered bits from the previous iteration are added to the beginning of the input bit stream prior to performing convolutional encoding. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output bit stream returns the convolutional-encoded code word. Wire this parameter to the <u>MT Convolutional Decode</u> VI to recover the input data stream.
- **final state** returns the final state for the k(K-1) shift registers as the right-aligned (least significant) k(K-1) bits, where K is the constraint length, and k is the data word length in bits.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain**

Warning from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
MT Convolutional Encoder (Rate)

This polymorphic instance generates an encoded bit stream based on a specified code rate. The code rate is equal to the ratio of the data word length to the code word length. This VI allows you to choose a code rate of 1/2, 1/3, 1/4, 2/3, or 3/4, using the **rate** parameter. The following table shows the available code rates and constraint lengths supported by this polymorphic instance. If you have a particular code rate/constraint length combination that is not available in the table, use the Generator Matrix instance to specify custom values.

Code Rate	Supported Constraint Lengths		
1/2	3,4,5,6,7,8,9,10,11,12,13,14,15,16,17		
1/3	3,4,5,6,7,8,9,10,11,12,13,14		
1/4	3,4,5,6,7,8,9,10,11,12,13,14		
2/3	2,3,4,5,6		
3/4	2,3,4		



- **input bit stream** specifies the bit sequence representing the data word to encode. Use the <u>MT Generate Bits (poly)</u> VI to produce this bit stream or wire a custom data bit stream to this parameter.
- **rate (k/n)** specifies the convolutional code rate as a ratio of k/n, where k is the input data word length and n is the output code word length. The default is 1/2.

1/2 (0)	The convolutional code rate is 1/2.
1/3 (1)	The convolutional code rate is 1/3.
1/4 (2)	The convolutional code rate is 1/4.
2/3 (3)	The convolutional code rate is 2/3.
3/4 (4)	The convolutional code rate is 3/4.

constraint length (K) specifies the maximum number of encoded bits that can be affected by a single input bit. This value represents

(1 + *maximal memory order*), where *maximal memory order* refers to the length of the longest shift register chain in the convolutional encoder. The default is 3.

- **initial state** specifies the encoder initialization state for the k(K-1) shift registers, where k is the input data word length and K specifies the constraint length. On the first call to this VI, and thereafter when **reset?** is set to TRUE (the default), the encoder state is reset to this value on each call to this VI. When **reset?** is set to FALSE, the final state from the previous iteration is used in subsequent calls to this VI and the **initial state** parameter is ignored. The default is 0.
- **TED** reset? specifies whether the internal state of the convolutional encoder is cleared. By default, the VI resets on first call. When reset? is set to TRUE, this VI clears any buffered bits from previous iterations, and the encoder is initialized to start from the specified initial state. When reset? is set to FALSE, this VI continues performing convolutional encoding from the previous iteration. Any buffered bits from the previous iteration are added to the beginning of the input bit stream prior to performing convolutional encoding. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more

information about the error.

- **(III)** output bit stream returns the convolutional-encoded code word. Wire this parameter to the <u>MT Convolutional Decode</u> VI to recover the input data stream.
- **final state** returns the final state for the k(K-1) shift registers as the right-aligned (least significant) k(K-1) bits, where K is the constraint length, and k is the data word length in bits.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The **input bit stream** array (*MessageData*) is divided into chunks of length equal to the data length (k). The convolutional encoder design sequentially encodes each chunk and recombines the encoded data to return the output code word in the **output bit stream**. The number of iterations required in the design to complete the encoding process is equal to: floor[length(*MessageData*)/k]

During the *i*th iteration, the chunk of *k* bits is represented by the *MessageData* vector defined by the following equation:

 $x_i = [x_i^0 \ x_i^1 \ \dots \ x_i^{k-2} \ x_i^{k-1}]^T$

The corresponding output encoded vector is represented by the following equation:

 $y_i = [y_i^0 \ y_i^1 \ \dots \ y_i^{n-2} \ y_i^{n-1}]^T$

The output encoded vector is generated from the input message vector using the binary connection polynomial matrix G. The transformation can be described by the equation:

in which the output encoded vector y is generated as the product of the binary connection polynomial matrix G and the input vector x.

Generator matrices corresponding to code rates and constraint lengths obtained from

Wicker, Stephen B. *Error Control Systems for Digital Communications and Storage*. Englewood Cliffs: Prentice-Hall, 1995.

MT Golay Encoder VI

Generates a Golay-encoded bit stream. The VI provides the two triplecorrecting Golay codes: the Golay (23,12,3) code and the extended Golay (24,12,3) code.



input bit stream specifies the bit sequence representing the data bits to encode. Use the <u>MT Generate Bits (poly)</u> VI to produce this bit stream or wire a custom data bit stream to this parameter.

- **Tip** To ensure that the **output bit stream** returned by the Golay encoder corresponds to the entire **input bit stream**, specify an **input bit stream** with an integer length that is a multiple of 12. If the length of the **input bit stream** is not an integer multiple of 12, the last {length(**input bit stream**) *modulo* 12} bits are buffered in the encoder. These bits are added to the beginning of the incoming bit stream during the beginning of the next iteration if **reset?** is set to FALSE.
- Golay code type (n,k,t) specifies the values of *n*, *k*, and *t* that define the code word length, data word length, and error correcting capacity, respectively, in bits.

Golay (23,12,3) (0)	The (n,k,t) values are (23, 12, 3). This is the default value.
Golay (24,12,3) (1)	The (n,k,t) values are (24, 12, 3).

- **reset?** specifies whether to clear the internal state of the encoder. By default, the VI resets on first call. When **reset?** is set to FALSE, the VI continues Golay encoding from the previous iteration. In this case, any buffered bits from the previous iteration are added to the beginning of the **input bit stream** prior to performing Golay encoding. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other

VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **[18]** output bit stream returns the Golay-encoded bits.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Hamming Encoder VI

Generates a Hamming-encoded bit stream.

The order-*m* Hamming codes are a special class of Bose-Chaudhari-Hocquenghem (BCH) codes in which the data word length equals 2^m -m-1, code word length equals 2^m -1 and the error correcting capacity equals 1. Valid values of *m* lie in the range $2 \le m \le 16$.



input bit stream specifies the bit sequence representing the data bits to encode. Use the <u>MT Generate Bits (poly)</u> VI to produce this bit stream or wire a custom data bit stream to this parameter.

- **Tip** To ensure that the **output bit stream** returned by the Hamming encoder corresponds to the entire **input bit stream**, specify an **input bit stream** with a length that is an integer multiple of $k = 2^m m 1$. If the length of the **input bit stream** is not an integer multiple of k, the last {length(**input bit stream**) *modulo k*} bits are buffered in the encoder. These bits are added to the beginning of the incoming bit stream during the beginning of the next iteration if **reset?** is set to FALSE.
- **hamming order** specifies the Hamming code order. Specify a value between 2–16, inclusive. The default is 3.
- **TEN reset?** specifies whether to clear the internal state of the encoder. By default, the VI resets on first call. When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the Hamming code. The VI also clears any buffered bits from previous iterations. When **reset?** is set to FALSE, the VI continues performing Hamming encoding from the previous iteration. Any buffered bits from the previous iteration are prepended to the **input bit stream** prior to encoding.
 - Tip When reset? is set to TRUE, the VI initializes the Galois field structure corresponding to the Hamming code.
 After initialization, this same Galois field structure is used for

performing the Hamming encoding provided **reset?** is set to FALSE. This initialization may consume significant amounts of time for code word lengths greater than 127. To improve the performance of the Hamming Encoder VI, NI recommends performing Hamming encoding with **reset?** set to TRUE for the first iteration and set to FALSE thereafter.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the Hamming-encoded bits.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel

error out indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Puncture Data Stream VI

Punctures the **input data stream** and returns an **output data stream** that is at a higher rate relative to the input.

Puncturing refers to the process of artificially increasing the rate of the input data by selectively deleting certain elements in the **input data stream**. The positions where the elements are deleted from the **input data stream** are defined by the **puncture pattern** matrix. Elements in the **input data stream** that correspond to zero values in the **puncture pattern** matrix are omitted from the **output data stream**. If the **input data stream** arrives at a rate R_{in} and the **puncture pattern** is a $P \times Q$ matrix comprising *L* zeros, the rate of the output data stream is given by

$$R_{out} = R_{in} * \frac{pq}{(pq - L)}$$

<u>Details</u>



- input data stream specifies the data stream with a rate to be increased by the process of puncturing. This data stream is usually obtained from the output of a block/convolutional encoder. The default is empty.
- **puncture pattern** specifies a binary-valued matrix that dictates which elements in the **input data stream** are omitted with respect to the **output data stream**. The default is 1.
 - Note The puncture pattern matrix is chronologically ordered by columns with respect to the **input data stream**. If the dimensions of the **puncture pattern** matrix are *m×n*, the *m* elements of the first column correspond to the first *m* elements of the **input data stream**, and so on.
- **reset?** specifies whether the VI uses the stored state information from its previous iteration while puncturing the **input data stream** in the current iteration. When **reset?** is set to TRUE, the VI clears all previous state information and reinitializes the buffers. The default is TRUE.

- Note When the length *L* of the **input data stream** is not an integer multiple of the product *P*×*Q* of the *P*×*Q* **puncture pattern** matrix, the last *L* modulo (*P*×*Q*) are buffered inside the VI. If **reset?** is set to FALSE, these buffered elements are prepended to the beginning of the **input data stream** during the next iteration, prior to the puncturing operation.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output data stream** returns the output data stream, which consists of all elements in the **input data stream** except those that were deleted during the puncturing operation.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for

more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

N

Code Rates R _{out}	Puncture Pattern	Output Data Stream
1/2	X:1	X1, Y1
	Y:1	
2/3	X:10	X1, Y1, Y2
	Y:11	
3/4	X:101	X1, Y1, Y2, X3
	Y:110	
5/6	X:10101	X1, Y1, Y2, X3, Y4, X5
	Y:11010	
7/8	X:1010101	X1, Y1, Y2, X3, Y4, X5, Y6, X7
	Y:1101010	

Typically, the puncturing operation occurs in conjunction with the channel encoding operation. In such a case, the number of rows *P* in the **puncture pattern** matrix should equal the code word length *n* of the mother block/convolutional code. The preceding table shows examples of puncturing matrices for deriving higher rate codes from a mother rate $\frac{1}{2}$ convolutional encoder. The **input data stream** is assumed to be of the form X1, Y1, X2, Y2, X3, Y3, and so on.

Note If you perform convolutional encoding followed by puncturing the data stream returned on the transmit side, you must depuncture the data prior to performing convolutional decoding on the receiving side. In addition, you must call the polymorphic instances of the convolutional decoder that implement either soft-decision or unquantized Viterbi decoding.

MT Reed Solomon Symbol Encode VI

Generates a normal or shortened Reed-Solomon (RS) encoded integer symbol stream.

Reed-Solomon codes are specialized Bose-Chaudhari-Hocquenghem (BCH) codes in which the code elements are drawn from the Galois field, $GF(2^m)$. Reed-Solomon codes are extremely powerful at burst error correction. The operations involved in Reed-Solomon encoding and decoding are performed in the $GF(2^m)$, meaning that the Reed-Solomon encoder operates on code words that are symbols/groups of bits. As such, the input parameters **code word length (n) symbols** and **data word length (k) symbols** are specified in terms of symbols. Each symbol represents a group of $Log_2(n+1)$ bits.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

<u>Details</u>

MT RS Symbol Encoder (normal)

This polymorphic instance generates a Reed-Solomon encoded integer symbol stream in normal format. The normal Reed-Solomon (n,k,t) code consists of values of n of the form $n=2^m-1$, where valid values of m lie in the range $2 \le m \le 16$.

For example, specify n=255, k=233 to produce a Reed-Solomon (255,233,11) code. This code has an error-correcting capacity of 11 Reed-Solomon symbols.



- modulation session specifies the modulation session refnum for configuring selected attributes of the Modulation Toolkit. Use this refnum to configure the advanced attributes in the MT Reed Solomon Symbol Encode VI. Initialize the session with the MT Open Session VI. After configuring the modulation session refnum with the appropriate attributes, wire the modulation session refnum into the RS Symbol Encoder VI. If the RS Symbol Encoder VI is called with an uninitialized modulation session refnum, the VI returns an error. The default is empty.
- input symbols specifies the array of integer symbols to be encoded using the Reed-Solomon encoding scheme. Use the MT <u>Generate Bits (poly)</u> VI followed by MT Pack Bits VI to generate the integer symbols required for this parameter.
 - Note The Reed-Solomon encoder is a block encoder, meaning that it operates on a block of *k* integers. To ensure that the **output symbols** stream corresponds to the entire **input symbols** array, specify an **input symbols** stream with a length *L* that is an integer multiple of *k* symbols. If the length *L* is not a multiple of *k* symbols, the last *L* modulo *k* symbols are buffered inside the encoder. These symbols are prepended to the incoming **input symbols** stream at the beginning of the next iteration if **reset?** is set to FALSE.

code word length (n), symbols specifies the length of the Reed-

Solomon code word. Each Reed-Solomon symbol represents $Log_2(n+1)$ bits. The default is 7.

- **data word length (k), symbols** specifies the length of the Reed-Solomon data word. Each Reed-Solomon symbol represents $Log_2(n+1)$ bits. The default is 3.
- **TEN reset?** specifies whether to clear the internal state of the Reed-Solomon encoder. By default, the VI resets on first call. When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the Reed-Solomon (*n*,*k*) values. The VI clears any buffered data from previous iterations. When **reset?** is set to FALSE, the VI continues performing Reed-Solomon encoding from the previous iteration. Any buffered data from the previous iteration are prepended to the **input symbols** stream prior to encoding. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output symbols** returns the Reed-Solomon encoded integer symbol stream. Wire this parameter to the <u>MT Reed Solomon</u> <u>Symbol Decode</u> VI to recover the input data stream.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT RS Symbol Encoder (shortened)

This polymorphic instance generates a Reed-Solomon encoded integer symbol stream in shortened format. Using shortened Reed-Solomon codes, the k and n values represent symbols. Each symbol has a length m bits, which is specified using the **bits per RS symbol** parameter.

Specify a shortened RS code by specifying the shortened values of *n* and *k*, as well as the bits per symbol that correspond to the underlying Galois field arithmetic. For example, you can specify **code word length (n) symbols** as 32, **data word length (k) symbols** as 28, and **bits per RS symbol** as 8 to generate a shortened Reed-Solomon (32,28,2) code with error-correcting capacity of two Reed-Solomon symbols. For this code, the underlying Galois field arithmetic is performed in $GF(2^8)=256$.



modulation session specifies the modulation session refnum for configuring selected attributes of the Modulation Toolkit. Use this refnum to configure the advanced attributes in the MT Reed Solomon Symbol Encode VI. Initialize the session with the MT Open Session VI. After configuring the modulation session refnum with the appropriate attributes, wire the modulation session refnum into the RS Symbol Encoder VI. If the RS Symbol Encoder VI is called with an uninitialized modulation session refnum, the VI returns an error. The default is empty.

- input symbols specifies the array of integer symbols to be encoded using the shortened Reed-Solomon encoding scheme. Use the <u>MT Generate Bits (poly</u>) VI followed by the <u>MT Pack Bits</u> VI to generate the integer symbols required for this control. The default is empty.
 - **Note** The shortened Reed-Solomon encoder is a block encoder, meaning that it operates on a block of *k* integers. To ensure that the **output symbols** stream corresponds to the entire **input symbols** array, specify an *input symbols* stream whose length *L* is an integer multiple of *k* symbols. If the

length *L* is not a multiple of *k* symbols, the last *L* modulo *k* symbols are buffered inside the encoder. These symbols are prepended to the incoming **input symbols** stream at the beginning of the next iteration if **reset?** is set to FALSE.

- **code word length (n), symbols** specifies the length of the Reed-Solomon code word. Each Reed-Solomon symbol represents $Log_2(n+1)$ bits. The default is 7.
- **data word length (k), symbols** specifies the length of the Reed-Solomon data word. Each Reed-Solomon symbol represents $Log_2(n+1)$ bits. The default is 3.
- bits per RS symbol specifies the number of bits per RS symbol. This parameter specifies the Galois field over which the encoding operation is performed as GF(2^{bits per RS symbol}). Valid values fall in the range 3–16 inclusive. For example, to specify a shortened Reed-Solomon (32,28) code in the Galois field GF(256), specify code word length (n) symbols as 32 symbols, code word length (k) symbols as 28 symbols, and bits per RS symbol as 8. The default is 3.
- **TEP reset?** specifies whether to clear the internal state of the Reed-Solomon encoder. By default, the VI resets on first call. When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the Reed-Solomon (*n*,*k*) values. The VI clears any buffered data from previous iterations. When **reset?** is set to FALSE, the VI continues performing Reed-Solomon encoding from the previous iteration. Any buffered data from the previous iteration are prepended to the **input symbols** stream prior to encoding. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain**

Error or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output symbols** returns the Reed-Solomon encoded integer symbol stream. Wire this parameter to the <u>MT Reed Solomon</u> <u>Symbol Decode</u> VI to recover the input data stream.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The following advanced attributes can be used to configure the behavior of the Reed-Solomon Symbol Encoder and the Reed-Solomon Symbol Decoder VIs. The property node is located on the Functions palette at **RF Communications**»Modulation»Digital.

🚆 🗝 Modulation Toolkit 💡	
RS Primitive Poly	۲
RS Symbol Mapping	۲
RS Galois primitive element	۲
RS Start Root Exp (Gen Poly)	7
RS Message Poly order	7
RS Shortening Zeros location	٢
RS Parity Poly order	٢
RS Parity Position	۲

- **RS Primitive Polynomial**: Use this attribute to set the coefficients of the primitive polynomial in the underlying Galois field GF(2^m). The coefficients of the primitive polynomial must be binary valued. The Reed-Solomon Symbol Encoder VI does not check whether the input polynomial is a primitive polynomial.
 - **Note** If the Primitive Polynomial array is not empty, the Reed-Solomon Symbol Encoder VI uses the specified polynomial. If the Primitive Polynomial array is empty, the Reed-Solomon Symbol Encoder VI employs a lookup table of stored primitive polynomials (for GF(2^m), m=2,...,16) for computing the Reed-Solomon generator polynomial.
- **RS Symbol Mapping**: Use this attribute to specify whether the input message and the output coded symbols are mapped to elements in the underlying Galois field GF(*n*+1) using the Polynomial representation or the Exponent representation.

Polynomial(default)	This format assumes that the message symbols lie in the range [0, <i>n</i>], corresponding to polynomials in GF(<i>n</i> +1).
Exponent	This format assumes that the input symbols represent Galois field exponents <i>i</i> , where i varies from –1 to <i>n</i> –1.

For example, consider the specification of a (7,3) Reed-Solomon double error-correcting code. The Reed-Solomon Symbol

Encoder VI can accept three Reed-Solomon message symbols (3 bits per symbol) and returns 7 Reed-Solomon encoded symbols. Assume that you input the message word [5 6 2] to the **input symbols** parameter of the Reed-Solomon Symbol Encoder VI. This example assumes that the **RS Message Polynomial order** property is set to **Descending Powers of X** and the **RS Parity Polynomial order** property is set to **Descending Powers of X**.

- **Case 1 RS Symbol Mapping** is set to polynomial. The Reed-Solomon encoder treats the array [5 6 2] as a polynomial $m(x)=(1+\alpha^2)x^2+(\alpha+\alpha^2)x+\alpha$. The Reed-Solomon parity polynomial is computed as $p(x)=x^2+\alpha^2 x+(1+\alpha)$ (or the array [0 1 4 3]) and returns the output encoded symbol array $c(x)=x^4 m(x)+p(x)$ [0 1 4 -1 3 5 6 2].
- **Case 2 RS Symbol Mapping** is set to Exponent. The Reed-Solomon encoder treats the array [5 6 2] as a polynomial $m(x)=_{\alpha} 5x^{2}+_{\alpha} 6x+_{\alpha} 2$. The Reed-Solomon parity polynomial is computed as $p(x)=x^{3}+_{\alpha} 4x^{2}+_{\alpha} 3$ (or the array [0 4 –1 3] and returns the output encoded symbol array $c(x)=x^{4} m(x)+p(x)[0 4 –1 3 5 6 2]$.
- **RS Galois primitive element**: Use this property to specify the exponent *q* of the primitive element $A=_{\alpha} q$ in the Galois field $GF(2^m)$. The element *q* determines the primitive element in the Galois field $GF(2^m)$. Valid values of *q* are such that GCD(q,2m-1)=1. The default value of *q* equals 1. The Reed-Solomon generator polynomial is uniquely specified by the primitive polynomial (mentioned previously), the Reed-Solomon start root exponent, and the Reed-Solomon Galois primitive element.

The Reed-Solomon generator polynomial g(x) is given as

 $g(x)=(x-A^B)(x-A^B+1)(x-A^{B+2})...(x-A^{B+2t-1}), A=a^4$

Here *B* denotes the Reed-Solomon start root exponent, *t* denotes the error-correcting capacity of the Reed-Solomon code, and $A=\alpha$ ⁴ denotes the Reed-Solomon Galois primitive element.

• **RS Start Root Exponent (Generator polynomial)**: Use this property to specify the starting root exponent of the Reed-

Solomon generator polynomial.

• **RS Message Polynomial order, RS Parity Polynomial order**: The Reed-Solomon Symbol Encoder/Decoder VIs interpret the input message stream/input coded stream as polynomials, with a format (increasing/decreasing powers of *X*) that is provided with attributes you set using the generic modulation session refnum. These attributes also determine the output order of the encoded symbols/recovered message symbols. The default value of both attributes is Decreasing Powers of *X*.

Decreasing Powers of X(default)	 This format treats the input message polynomial in decreasing powers of <i>x</i>. Note The code-word polynomial is also output in the same order.
	For example, consider the specification of a (7,3) Reed-Solomon double error-correcting code. The Reed-Solomon encoder inputs three Reed- Solomon message symbols (3 bits per symbol) and generates 7 Reed-Solomon encoded symbols, assuming that you enter the message word [5 6 2] to the input symbols parameter of the Reed- Solomon encoder.
	If RS Message Polynomial Order is set to Decreasing Powers of <i>X</i> , and the RS Parity Polynomial order is set to Decreasing Powers of <i>X</i> , the Reed-Solomon Symbol Encoder VI treats the array as a polynomial $m(x)=(1+\alpha^2)x^2+(\alpha+\alpha^2)x+\alpha$. The parity polynomial is computed to be $p(x)=x^2+\alpha^2$ $x+(1+\alpha)$ and returns the output encoded symbol array to be $c(x)=x^4 m(x)+p(x)$ [0 1 4 3 5 6 2] (if RS Parity Position is set to Beginning) or $c(x)=m(x)+x^4 p(x)$ [5 6 2 0 1 4 3] (if RS Parity Position is set to End).
Increasing Powers of X	This format treats the input message polynomial in increasing powers of <i>x</i> . If RS Message Polynomial Order is set to

Increasing Powers of *X*, and **RS Parity Polynomial** order is set to Increasing Powers of *X*, the Reed-Solomon Symbol Encoder VI treats the array as a polynomial $m(x)=(1+\alpha^2)+(\alpha+\alpha^2)x+\alpha x^2$ and returns the output encoded symbol array to be [2 5 1 6 5 6 2] (if **RS Parity Position** is set to Beginning) or [5 6 2 2 5 1 6] (if **RS Parity Position** is set to End).

• **RS Shortening Zeros Location**: Use this property to specify the relative position of the shortening zeros with respect to the message polynomial prior to performing Reed-Solomon encoding.

Prior Message(default)	Places the shortening zeros before the message bits.
Post Message	Places the shortening zeros after the message bits.

• **RS Parity Position**: Use this property to specify the position of the parity symbols relative to the message symbols in the Reed-Solomon code word.

Beginning(default)	Places the parity bits at the beginning of the code word.
End	places the parity bits at the end of the code word.

The values of (n,k) passed to the RS decoder are specified in terms of RS symbols, where *n* denotes the RS code word length (symbols) and *k* denotes the RS data word length (symbols). For decoding, choose either a normal RS code or a shortened RS code. Each normal RS-encoded symbol corresponds to $Log_2(n + 1)$ bits. Each shortened RS-encoded symbol corresponds to the number of bits specified in the **bits per RS symbol** parameter.

The following table relates RS code types to data and word lengths, where L is the number of shortening zeros.

Code Type	Bits per RS Symbol (<i>m</i>)	Code Word Length (<i>n</i>)	Data Word Length (<i>k</i>)	Error- Correcting Capacity (<i>t</i>)

Reed- Solomon (normal)	Log ₂ (n+ 1)	2 ^m - 1	2 ^m - 1 - 2t	(n - k)/2
Reed- Solomon (shortened)	User- specified	2m - 1 - L	2 ^m - 1 - L - 2t	(n - k)/2

MT Spread Symbols VI

Performs the <u>DSSS</u> spreading operation given a user-defined spreading code. The spreading code algorithm performs non-return-to-zero (NRZ) encoding of the **input bit stream** and the **spreading code**. However, the output is an array of zeros and ones.

<u>Details</u>



- **input bit stream** specifies the sequence of information bits to be spread.
- **spreading code** specifies the sequence of bits that serve as the spreading code.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output chip stream returns the spread sequence of data bits for transmission. The number of elements in this array equals the product of the input bit stream array length and the spreading

code array length.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Each bit in the **input bit stream** is spread according to the following table.

Spreading Input	Spreading Output
0	spreading code
1	complement of spreading code

For example, an **input bit stream** of 1100 and a **spreading code** input of 1010 return an **output chip stream** of 0101 0101 1010 1010.

Interleaving

Owning Palette: Digital

Use the VIs on the **Interleaving** palette to perform permutation, and block and convolutional interleaving.

Palette Object	Description
<u>MT Block</u> Interleave (Matrix)	Performs interleaving by writing the input data into an m (rows) × n (columns) matrix, row by row, and then reading the data out column by column when the matrix is full. Typically, block interleaver and deinterleaver VIs are used together.
MT Permute	Changes the position of input data elements based on values specified in the permutation array parameter.
<u>MT</u> Convolutional Interleave	Performs the interleaving process using a set of branches that the data elements cyclically pass through.

MT Block Interleave (Matrix) VI

Performs interleaving by writing the input data into an m (rows) $\times n$ (columns) matrix, row by row, and then reading the data out column by column when the matrix is full. Typically, block interleaver and deinterleaver VIs are used together.

<u>Details</u>



- **data in** specifies the input data to be interleaved. Data is processed in blocks of *mn*, where *m* is the number of rows and *n* is the number of columns. If the number of data samples is less than block size *mn*, the block interleaver does not generate any data if the VI is run once. If the data samples are continuously fed in and if the number of data samples is greater than but not a multiple of *mn*, the block interleaver buffers the remaining samples for use in the next iteration. For example, if *m*=3, *n*=4, and 27 data samples are specified, the VI interleaves two blocks ($2 \times 12 = 24$) of data and buffers 3 samples (27-24=3) for use in the next iteration. Data is read in by row and read out by column. The default is empty.
- **number of rows** specifies the number of rows of the block interleaver. The default is 1.
- **number of columns** specifies the number of columns of the block interleaver. The default is 1.
- **TEP** reset? specifies how the VI handles buffered data. When reset? is set to TRUE, buffered data is cleared. Input parameters are checked on a first call to this VI. Any change in the input parameter values during subsequent iterations of the VI is reflected only if reset? is set to TRUE. In continuous operations, when the buffered data is appended in the beginning of data from next iteration, set reset? to FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain

Error or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **[132]** data out returns the output of the block interleave operation.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Refer to the following theoretical example for an illustration of the interleave operation.

If the **data in** parameter is entered as 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14, ..., with **number of rows** equal to 3 and **number of columns** equal to 5, the block could be visualized as follows:

0	1	2	3	4
5	6	7	8	9
10	11	12	13	14

The data is entered by row and read by column, such that, for this example, **data out** would be 0,5,10,1,6,11,2,7,12,3,8,13,4,9,14.

MT Convolutional Interleave VI

Performs the interleaving process using a set of branches that the data elements cyclically pass through.

Each branch has different delays associated with it. Hence the data input of each interleaver branch is delayed by a specific amount (the amount of delay in that particular branch) before the interleaver returns the data. A convolutional interleaver is twice as efficient as a block interleaver.

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Convolutional Interleaver (Standard)

This polymorphic instance performs the interleaving process using a fixed number of branches and a fixed unit delay.



- **data in** specifies the input data to the interleaver. The default is empty.
- number of branches specifies the number of branches of the convolutional interleaver. Data elements pass through the branches in a cyclic fashion. For example, in an *N* branch convolutional interleaver, data element 0 goes through branch 0, element 1 goes through branch 1, element *N*–1 goes through branch *N*–1, element *N* returns through branch 0 and so on. Each branch incorporates different delays. The default is empty.
- **unit delay** specifies the unit delay value. If this value is defined as D, then the number of delays on the *i*th branch is (*i*×*D*). If the total number of branches is *N*, then *i*=0,1,..., *N*–1. The default is 1.
- **initial state** specifies the shift register values when the convolutional interleaver begins operation. The default is empty.
- **reset?** specifies whether to check the current input parameters. On the first call to this VI or when **reset?** is set to TRUE, the VI checks input parameters. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **[132]** data out returns the convolutional interleaved data.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
MT Convolutional Interleaver (Generic)

This polymorphic instance performs the interleaving process with userspecified delays for each branch.



- **data in** specifies the input data to the interleaver. The default is empty.
- **interleaver delays** specifies the number of delays individually for each branch of the corresponding interleaver. The number of elements in this array determines the number of branches, and the value of each element denotes the delay for that branch. For example, if **interleaver delays** is [3,6,9,2], then it implies that branch 0 has 3 delays, branch 1 has 6 delays, and so on, for the corresponding interleaver. The default is empty.



Note You must specify a value for this parameter.

- **initial state** specifies the shift register values when the convolutional interleaver begins operation. The default is empty.
- **reset?** specifies whether to check the current input parameters. On the first call to this VI or when **reset?** is set to TRUE, the VI checks input parameters. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front

panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **[132]** data out returns the convolutional interleaved data.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The following example demonstrates convolutional interleaving/deinterleaving.

Let the **data in** be: x0, x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, x11, x12, x13, ...



D = unit delay in the path.

Assume for this example that the unit delay = 1, and the initial state shift registers are initialized with values of 0 for both the interleaver and the deinterleaver.

Interleaved Data: x0, 0, 0, 0, x4, x1, 0, 0, x8, x5, x2, 0, x12, x9,



References

- 1. Wicker, Stephen B. *Error Control Systems for Digital Communication and Storage*. Englewood Cliffs: Prentice-Hall, 1995.
- 2. Proakis, John G. and Masoud Salehi *Communication Systems Engineering.* 2d ed. Englewood Cliffs: Prentice-Hall, 2001.
- 3. Sklar, Bernard. *Digital Communications: Fundamentals and Applications.* 2d ed. Englewood Cliffs: Prentice-Hall, 2001.

MT Permute VI

Changes the position of input data elements based on values specified in the **permutation array** parameter.

Details



- **data in** specifies input data to be permuted. The data samples are rearranged into new locations as defined by the permutation array. The default is empty.
- **permutation array** specifies an array whose elements define the location of the samples in the output data. Refer to the details section for more information about how this VI handles permutation operations. The default is empty.
- **reset?** specifies how the VI handles buffered data. When **reset?** is set to TRUE, buffered data is cleared. Input parameters are checked on a first call to this VI. Any change in the values of the input parameters during subsequent iterations of the VI is reflected only if **reset?** is set to TRUE. In continuous operations, when the buffered data is prepended to the beginning of data from next iteration, set **reset?** to FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about

the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **data out** returns the permuted data. The permuted data is typically passed through <u>MT DePermute</u> VI to retrieve the original data.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Refer to the following theoretical example for an illustration of the permutation operation.

If the **data in** array is entered as 7,13,17,9,2, and the **permutation array** is 3,2,4,0,1, the **data out** array would be 9,21,13,7,17, as illustrated in the following figure:



Modulation

Owning Palette: Digital

Use the VIs on the **Modulation** palette to implement several types of digital modulation.

Palette Object	Description
<u>MT</u>	Receives a sequence of data bits, performs <u>ASK modulation,</u>
<u>Modulate</u>	and returns the modulated complex <u>baseband</u> waveform in
<u>ASK</u>	the output complex waveform parameter.
<u>MT</u>	Receives a sequence of data bits, performs PAM <u>modulation</u> ,
<u>Modulate</u>	and returns the modulated complex <u>baseband</u> waveform in
<u>PAM</u>	the output complex waveform parameter.
<u>MT</u>	Receives a sequence of data bits, performs QAM modulation,
<u>Modulate</u>	and returns the modulated complex baseband waveform in
QAM	the output complex waveform parameter.
<u>MT</u>	Receives a sequence of data bits, performs <u>FSK modulation</u> ,
<u>Modulate</u>	and returns the modulated complex <u>baseband</u> waveform in
FSK	the output complex waveform parameter.
<u>MT</u> <u>Modulate</u> <u>MSK</u>	Receives a sequence of data bits, performs <u>MSK modulation</u> , and returns the modulated complex <u>baseband</u> waveform in the output complex waveform indicator.
<u>MT</u>	Receives a sequence of data bits, performs <u>PSK modulation</u> ,
Modulate	and returns the modulated complex <u>baseband</u> waveform in
PSK	the output complex waveform parameter.
MT	Receives a sequence of data bits, performs <u>CPM modulation</u> ,
Modulate	and returns the modulated complex <u>baseband</u> waveform in
CPM	the output complex waveform parameter.

MT Modulate ASK VI

Receives a sequence of data bits, performs <u>ASK modulation</u>, and returns the modulated complex <u>baseband</u> waveform in the **output complex waveform** parameter.

ASK modulation involves mapping the **input bit stream** to ASK symbols (as specified by the ASK **symbol map**), interpolating by zero-insertion, and applying the pulse-shaping filter to the interpolated I/Q data.

Note Using FIR pulse-shaping filters in modulation VIs may lead to the apparent loss of bits caused by filter delay. Refer to Filter Delay for more information about this effect. Set the flush buffers? parameter to TRUE to offset the effects of filter delay in single-shot operations or in the last iteration of continuous operations.

ASK system parameters symbol rate (Hz) pulse shaping filter coeffi reset? (T) flush buffers? (F)
--

input bit stream specifies the sequence of information bits to be modulated.



Note For ASK systems with more than 1 bit per symbol, such as 4-ASK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the input bits are 2,1,... the generated symbols correspond to 0,1,1,0,...

ASK system parameters specifies parameter values defining the ASK system. Wire the ASK system parameters cluster returned by the ASK instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of ASK levels in the array is 2^N, where N is the number of bits per symbol. The

length of the vector for the symbols farthest from the origin is 1.

- **symbol rate** specifies the desired symbol rate, in hertz. This value must be positive. The default is 1.00.
- **pulse shaping filter coefficients** specifies an ordered array containing the desired sample-spaced coefficients for the pulseshaping filter. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **ASK system parameters** cluster, which is passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the modulated signal, and the returned output data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- **reset?** specifies whether the VI continues modulating using the previous iteration states. The VI resets on the first call and when **reset?** is set to TRUE. **reset?** must be set to TRUE the first time this VI is called and whenever you want to restart the modulator. The default is TRUE.
- **flush buffers?** forces out samples from the modulated waveform that are affected by the FIR pulse-shaping filter delay. Set this parameter to TRUE during single-shot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the ASK-modulated complex baseband waveform data.
 - **ID** returns the time of first value in **Y** array.
 - **bell** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbols out** returns the array of mapped symbols before pulse shaping is applied. The array represents the complex value of each mapped symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select

Explain Error or **Explain Warning** from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Modulate CPM VI

Receives a sequence of data bits, performs <u>CPM modulation</u>, and returns the modulated complex <u>baseband</u> waveform in the **output complex waveform** parameter.

		-
- 1	Υ.	1
1		٠,
	-	,

Note Using FIR pulse-shaping filters in modulation VIs may lead to the apparent loss of bits caused by filter delay. Refer to <u>Filter</u> <u>Delay</u> for more information about this effect. Set the **flush buffers?** parameter to TRUE to offset the effects of filter delay in single-shot operations or in the last iteration of continuous operations.

Details



input bit stream specifies the sequence of information bits to be modulated.

 $\overline{\mathbb{N}}$

Note For CPM systems with more than 1 bit per symbol, such as 4-CPM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...

CPM system parameters specifies parameter values defining the CPM system. Wire the CPM system parameters cluster from the CPM (M) or CPM (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies the number of samples per symbol in the modulated output complex waveform. It should be an even number.
- **symbol rate** specifies the desired symbol rate, in hertz. This value must be positive. The default is 1.00.

- **pulse shaping filter coefficients** specifies an ordered array containing the desired pulse-shaping coefficients. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter</u> <u>Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the **CPM system parameters** cluster passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the modulated signal, and the returned output data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to prime the filter, hence eliminating the transient.
- **reset?** specifies whether the VI continues modulation using the previous iteration states. The VI resets on the first call and when **reset?** is set to TRUE. **reset?** must be set to TRUE whenever you want to restart the modulator.
- **flush buffers?** forces out samples from the modulated waveform that are affected by the FIR pulse-shaping filter delay. Set this parameter to TRUE during single-shot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain**

Error or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the CPM-modulated complex baseband waveform data.
 - **ID** returns the time of first value in **Y** array.
 - **DEL** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbols out** returns the array of mapped symbols, before pulse shaping is applied. The real portion corresponds to the carrier frequency offset of the mapped symbol, and the imaginary portion is zero.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The bandwidth of a CPM signal, using Carson's Rule, is approximately given by the following formula:

BW = 2(Fm + Fd)

where

Fm is symbol bandwidth and is given by $(1 + \alpha) * R$, where *R* is the symbol rate and α is the filter parameter

 $\max(h_i) \times M \times R$

Fd is the maximum frequency deviation and is given by 2^{2} where h_{i} is the modulation index for the *i*th baud

The sample rate of the signal is given by fs = samples per symbol * R.

A value of *f*s that is less than *BW* causes aliasing. Refer to the *Aliasing* topic in the *LabVIEW Help* for more information about aliasing.

You can prevent your signal from aliasing by using one of the following strategies:

- Increasing the sample rate by increasing the samples per symbol.
- Decreasing the bandwidth of the message signal by decreasing the filter parameter α .
- Decreasing the maximum modulation index (h_i) .

MT Modulate FSK VI

Receives a sequence of data bits, performs <u>FSK modulation</u>, and returns the modulated complex <u>baseband</u> waveform in the **output complex waveform** parameter.

	-	-	40		
2	-		Т		
1		1	ſ	1	
í	۱,	1	1	1	
	٠,	1		1	

Note Using FIR pulse-shaping filters in modulation VIs may lead to the apparent loss of bits caused by filter delay. Refer to <u>Filter</u> <u>Delay</u> for more information about this effect. Set the **flush buffers?** parameter to TRUE to offset the effects of filter delay in single-shot operations or in the last iteration of continuous operations.

Details



input bit stream specifies the sequence of information bits to be modulated.

 $\mathbf{\nabla}$

Note For FSK systems with more than 1 bit per symbol, such as 4-FSK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...

FSK system parameters specifies parameter values defining the FSK system. Wire the **FSK system parameters** cluster returned by the FSK (M) or FSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of FSK levels in the array must be 2^{*N*}, where *N* is the number of bits per symbol.
- symbol phase continuity specifies whether the phase

transitions between symbols are continuous.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, discontinuous phase FSK (DPFSK).

- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.
- **pulse shaping filter coefficients** specifies an ordered array containing the desired pulse-shaping coefficients. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter</u> <u>Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **FSK system parameters** cluster passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the modulated signal, and the returned output data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- **reset?** specifies whether the VI continues modulation using the previous iteration states. The VI resets on the first call and when **reset?** is set to TRUE. **reset?** must be set to TRUE whenever you want to restart the modulator. The default is TRUE.
- **flush buffers?** forces out samples from the modulated waveform that are affected by the FIR pulse-shaping filter delay. Set this parameter to TRUE during single-shot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set

reset? to TRUE on the subsequent iteration.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the FSK-modulated complex baseband waveform data.
 - **ID** returns the time of first value in **Y** array.
 - **DEED** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbols out** returns the array of mapped symbols, before pulse shaping is applied. The real portion corresponds to the carrier frequency offset of the mapped symbol, and the imaginary portion is zero.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and

select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The bandwidth of a CPM signal, using Carson's Rule, is approximately given by the following formula:

BW = 2(Fm + Fd)

where

Fm is symbol bandwidth and is given by $(1 + \alpha) * R$, where *R* is the symbol rate and α is the filter parameter

Fd is the maximum frequency deviation

The sample rate of the signal is given by fs = samples per symbol * R.

A value of *f*s that is less than *BW* causes aliasing. Refer to the *Aliasing* topic in the *LabVIEW Help* for more information about aliasing.

The signal can be checked for aliasing by observing its spectrum and noting whether the observed bandwidth of the signal is higher than the calculated bandwidth.

You can prevent your signal from aliasing by using one of the following strategies:

- Increasing the sample rate by increasing the samples per symbol.
- Decreasing the bandwidth of the message signal by decreasing the filter parameter α .
- Decreasing the maximum modulation index (h_i) .

MT Modulate MSK VI

Receives a sequence of data bits, performs <u>MSK modulation</u>, and returns the modulated complex <u>baseband</u> waveform in the **output complex waveform** indicator.

	2	-		
	R	þ	h	5
1		٩	1	l
۰.	-	_	1	l

Note Using FIR pulse-shaping filters in modulation VIs may lead to the apparent loss of bits caused by filter delay. Refer to Filter Delay for more information about this effect. Set the **flush buffers?** parameter to TRUE to offset the effects of filter delay in single-shot operations or in the last iteration of continuous operations.



- **input bit stream** specifies the sequence of information bits to be modulated.
- **MSK system parameters** specifies parameter values defining the MSK system. Wire the **MSK system parameters** cluster returned by the MSK instance of the <u>MT Generate System Parameters</u> VI to this cluster. Do not alter the values.
 - differential encoding enable specifies whether the bit sequence is differentially encoded.

disable (0)	Does not allow bit sequence encoding.
enable (1)	Allows bit sequence encoding. This is the default value.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.
- **pulse shaping filter coefficients** specifies an ordered array containing the desired pulse-shaping coefficients. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter</u>

<u>Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **MSK system parameters** cluster passed to this VI.

- Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the modulated signal, and the returned output data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- **reset?** specifies whether the VI continues modulating using the previous iteration states. The VI resets on the first call and when **reset?** is set to TRUE. **reset?** must be set to TRUE the first time this VI is called and whenever you want to restart the modulator. The default is TRUE.
- **flush buffers?** forces out samples from the modulated waveform that are affected by the FIR pulse-shaping filter delay. Set this parameter to TRUE during single-shot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the MSK-modulated complex baseband waveform data.
 - **ID** returns the time of first value in **Y** array.
 - **Description between values in the Y array.**
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbols out** returns the array of mapped symbols, before pulse shaping is applied. The real portion corresponds to the carrier frequency offset of the mapped symbol, and the imaginary portion is zero.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.

MT Modulate PAM VI

Receives a sequence of data bits, performs PAM <u>modulation</u>, and returns the modulated complex <u>baseband</u> waveform in the **output complex waveform** parameter.

PAM modulation involves mapping the input bit stream to PAM symbols (as specified by the PAM **symbol map**), interpolating by zero-insertion, and applying the pulse-shaping filter to the interpolated I/Q data.

Note Using FIR pulse-shaping filters in modulation VIs may lead to the apparent loss of bits caused by filter delay. Refer to Filter
Delay for more information about this effect. Set the flush
buffers? parameter to TRUE to offset the effects of filter delay in single-shot operations or in the last iteration of continuous operations.

Input bit stream PAM system parameters symbol rate (Hz) pulse shaping filter coeffi error in (no error) reset? (T) flush buffers? (F)

input bit stream specifies the sequence of information bits to be modulated.



M

Note For PAM systems with more than 1 bit per symbol, such as 4-PAM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the input bits are 1,0,0,1,... the generated symbols correspond to 0,1,1,0,...

PAM system parameters specifies parameter values defining the PAM system. Wire the **PAM system parameters** cluster returned by the PAM (M) or PAM (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of PAM levels in the array is 2^N, where N is the number of bits per symbol. The

vector length for the symbol(s) farthest from the origin is 1.

- **symbol rate** specifies the desired symbol rate, in hertz. This value must be positive. The default is 1.00.
- **pulse shaping filter coefficients** specifies an ordered array containing the desired sample-spaced coefficients for the pulseshaping filter. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **PAM system parameters** cluster, which is passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the modulated signal, and the returned output data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- **reset?** specifies whether the VI continues modulating using the previous iteration states. The VI resets on the first call and when **reset?** is set to TRUE. **reset?** must be set to TRUE the first time this VI is called and whenever you want to restart the modulator. The default is TRUE.
- **flush buffers?** forces out samples from the modulated waveform that are affected by the FIR pulse-shaping filter delay. Set this parameter to TRUE during single-shot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other

VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the PAM-modulated complex baseband waveform data.
 - **ID** returns the time of first value in **Y** array.
 - **bell** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbols out** returns the array of mapped symbols before pulse shaping is applied. The array represents the complex value of each mapped symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for

more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Modulate PSK VI

Receives a sequence of data bits, performs <u>PSK modulation</u>, and returns the modulated complex <u>baseband</u> waveform in the **output complex waveform** parameter.

	2	-		
	R	þ	h	5
1		٩	1	l
۰.	-	_	1	l

Note Using FIR pulse-shaping filters in modulation VIs may lead to the apparent loss of bits caused by filter delay. Refer to Filter Delay for more information about this effect. Set the **flush buffers?** parameter to TRUE to offset the effects of filter delay in single-shot operations or in the last iteration of continuous operations.



- **input bit stream** specifies the sequence of information bits to be modulated.
 - **Note** For PSK systems with more than 1 bit per symbol, such as 4-PSK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...

PSK system parameters specifies parameter values defining the PSK system. Wire the PSK system parameters cluster from the PSK (M) or PSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.
- **differential PSK** specifies the status of the differential PSK.

disable (0) Does not allow bit sequence encoding.

enable (1)	Allows bit sec	quence encoding.
------------	----------------	------------------

PSK type specifies the type of PSK modulation.

-	
normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase- shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by $1/2$ of a symbol period from the ideal symbol timing for I. offset is currently only supported for $M = 4$.

- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.
- **pulse shaping filter coefficients** specifies an ordered array containing the desired pulse-shaping coefficients. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter</u> <u>Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **PSK system parameters** cluster which is passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the modulated signal, and the returned output data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.

reset? specifies whether the VI continues modulating using the previous iteration states. The VI resets on the first call and when **reset?** is set to TRUE. **reset?** must be set to TRUE the first time this VI is called and whenever you want to restart the modulator. The default is TRUE.

- **flush buffers?** forces out samples from the modulated waveform that are affected by the FIR pulse-shaping filter delay. Set this parameter to TRUE during single-shot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the PSK-modulated complex baseband waveform data.
 - **ID** returns the time of first value in **Y** array.
 - **besilt of the time between values in the Y array.**
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (]) and

quadrature-phase (Q) data, respectively.

- **symbols out** returns the array of mapped symbols before pulse shaping is applied. The array represents the complex value of each mapped symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Modulate QAM VI

Receives a sequence of data bits, performs <u>QAM</u> <u>modulation</u>, and returns the modulated complex <u>baseband</u> waveform in the **output complex waveform** parameter.



Note Using FIR pulse-shaping filters in modulation VIs may lead to the apparent loss of bits caused by filter delay. Refer to <u>Filter</u> Delay for more information about this effect. Set the **flush buffers?** parameter to TRUE to offset the effects of filter delay in single-shot operations or in the last iteration of continuous operations.



- **input bit stream** specifies the sequence of information bits to be modulated.
 - **Note** For QAM systems with more than 1 bit per symbol, such as 4-QAM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...

QAM system parameters specifies parameter values defining the QAM system. Wire the QAM system parameters cluster returned by the QAM (M) or QAM (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - 2

Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

symbol map specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^N, where N

is the number of bits per symbol. The length of the vector for the symbol(s) farthest from the origin must be 1.

- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.
- **pulse shaping filter coefficients** specifies an ordered array containing the desired pulse-shaping coefficients. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter</u> <u>Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **QAM system parameters** cluster, which is passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the modulated signal, and the returned output data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- **reset?** specifies whether the VI continues modulating using the previous iteration states. The VI resets on the first call and when **reset?** is set to TRUE. **reset?** must be set to TRUE the first time this VI is called and whenever you want to restart the modulator. The default is TRUE.
- **flush buffers?** forces out samples from the modulated waveform that are affected by the FIR pulse-shaping filter delay. Set this parameter to TRUE during single-shot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other

VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the QAM-modulated complex baseband waveform data.
 - **ID** returns the time of first value in **Y** array.
 - **bell** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbols out** returns the array of mapped symbols before pulse shaping is applied. The array represents the complex value of each mapped symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for
more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Decoding

Owning Palette: Digital

Use the VIs on the **Decoding** palette to decode, despread, and depuncture baseband data streams.

Palette Object	Description
MT Despread Symbols	Performs the <u>direct sequence spread spectrum</u> (DSSS) despreading operation on a spread sequence of data bits.
MT Convolutional Decode	Decodes a convolutionally encoded bit stream, using a specified rate or generator matrix. The decoding scheme employs the Viterbi decoding algorithm and returns an output binary message stream.
MT Reed Solomon Symbol Decode	Decodes a normal or shortened Reed-Solomon (RS) encoded integer symbol stream.
<u>MT Golay</u> <u>Decoder</u>	Decodes an <u>Golay-encoded</u> bit stream. The decoder provides the two triple-correcting Golay codes: the Golay (23,12,3) code and the extended Golay (24,12,3) code. This VI uses the Arithmetic decoding algorithm for decoding the Golay (23,12,3) encoded bit stream and the Kasami error-trapping decoding algorithm for decoding the extended Golay (24,12,3) encoded bit stream.
<u>MT Hamming</u> Decoder	Decodes a Hamming-encoded bit stream.
MT BCH Decoder	Performs decoding on a Bose-Chaudhari-Hocquenghem (BCH)-encoded bit stream using the Berlekamp algorithm for binary BCH codes. The input values of n , k , and t define the code word length, data word length and error correcting capacity, respectively, in bits. The VI provides a set of common combinations of (n,k,t).
MT Depuncture Data Stream	Inserts an erasure value inside the punctured input data stream . The erasure value is equivalent to inserting a null symbol in the position of the input data stream that was previously punctured using the <u>MT Puncture Data</u>

	Stream VI.
<u>MT LDPC</u> <u>Decoder</u>	Message-passing algorithms are effective decoding strategies for <u>low-density parity check</u> (LDPC). The best algorithm known is iterative probabilistic decoding, or belief propagation. In this algorithm, at each iteration, messages are passed from message nodes to check nodes and vice versa. The message that is sent from message node (v) to the check node (c) must not take into account the message sent in the previous iteration from c to v . The same is true for messages passed from check nodes to message nodes.

MT BCH Decoder VI

Performs decoding on a Bose-Chaudhari-Hocquenghem (BCH)-encoded bit stream using the Berlekamp algorithm for binary BCH codes. The input values of \mathbf{n} , \mathbf{k} , and \mathbf{t} define the code word length, data word length and error correcting capacity, respectively, in bits. The VI provides a set of common combinations of (n,k,t).

<u>Details</u>



- **input bit stream** specifies the bit sequence representing the code word to decode. You can use the <u>MT BCH Encoder</u> VI to generate the encoded bit stream. The default is empty.
- **n,k,t** specifies the values of *n*, *k*, and *t* that define the code word length, data word length, and error-correcting capacity, respectively, in bits. The *n*, *k*, and *t* values for BCH encoding are interrelated such that specifying two values constrains the third value. Select from a set of well-known (n,k,t) triples that are widely used for BCH encoding or specify an arbitrary triple by entering three commaseparated integers. The default is 7, 4, 1.
- **TEN reset?** specifies whether to clear the internal state of the decoder. By default, the VI resets on first call. When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the BCH (n,k,t) values. The VI clears any buffered bits from previous iterations. When **reset?** is set to FALSE, the VI continues performing BCH decoding from the previous iteration. Any buffered bits from the previous iteration are added to the beginning of the **input bit stream** prior to performing BCH decoding. The default is TRUE.
 - **Tip** When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the user-specified (n,k,t) combination. Once initialized, this same Galois field structure is used for performing the BCH decoding provided **reset?** is set to FALSE. This initialization may consume significant amounts of time for values of *n* greater than 127.

Therefore, to get the best performance from the BCH decoder, NI recommends you perform BCH decoding with **reset?** set to TRUE for the first iteration, and set to FALSE thereafter.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **[18]** output bit stream returns the decoded bit stream.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about

the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The prepopulated (n, k, t) parameter contains all valid codes for values of n up to 511. However, the Modulation Toolkit supports BCH encoding and decoding up to Galois field sizes of GF(65536), so you can specify BCH code word lengths up to 65535. Do so by entering valid values of the code word length (n), data word length (k) and error correction capacity (t) separated by commas inside the string combo box.

Note Not all (n, k, t) combinations specify valid BCH codes. The BCH encoder/decoder VIs perform validation for values of the code-word length (*n*). If *n* is not of the form 2^m –1, the VI returns an error. The only validation criterion for the data-word length (*k*) and error-correcting capacity (*t*) is that they should be positive integers strictly lesser than *n*. The Modulation Toolkit does not guarantee specifications of BCH (*n*,*k*,*t*) values that satisfy the above criterion, but are not valid BCH codes.

For details on the Berlekamp's algorithm for binary BCH decoding, refer to the following sources:

Lin, S. and DJ Costello, Jr. *Error Control Coding: Fundamentals and Applications.* Englewood Cliffs: Prentice-Hall, 1983.

Wicker, Stephen B. *Error Control Systems for Digital Communications and Storage.* Englewood Cliffs: Prentice-Hall, 1995.

MT Convolutional Decode VI

Decodes a convolutionally encoded bit stream, using a specified rate or generator matrix. The decoding scheme employs the Viterbi decoding algorithm and returns an output binary message stream.

The Modulation Toolkit implements the convolutional decoding algorithm as a polymorphic VI with polymorphic instances corresponding to different types of the Viterbi decoding algorithm (hard decision/unquantized/soft decision) and the type of specification (code rate/generator matrix).

•

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Convolutional Decoder (Viterbi Hard Decision, Rate)

This polymorphic instance decodes a convolutionally encoded bit stream using a specified code rate. The code rate is equal to the ratio of the data word length to the code word length.

The following table shows the available code rates and constraint lengths supported by this polymorphic instance. If you have a particular code rate/constraint length that is not available in the table, use a generator matrix instance of this VI to specify custom values.

Code Rate	Supported Constraint Lengths
1/2	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17
1/3	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14
1/4	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14
2/3	2, 3, 4, 5, 6
3/4	2, 3, 4



- **input bit stream** specifies the bit sequence representing the code word to decode. You can use the <u>MT Convolutional Encode</u> VI to generate the encoded bit stream. The default is empty.
- **rate (k/n)** specifies the convolutional code rate as a ratio *k/n*, where *k* is the input data word length and *n* is the output code word length. Use a generator matrix instance of this VI to specify a different rate. The default is 1/2.

1/2 (0)	The convolutional code rate is 1/2.
1/3 (1	The convolutional code rate is 1/3.
1/4 (2	The convolutional code rate is 1/4.
2/3 (3	The convolutional code rate is 2/3.
3/4 (4	The convolutional code rate is 3/4.

Constraint length (K) specifies the maximum number of encoded

bits that can be affected by a single input bit. This value represents (1 + maximal memory order), where maximal memory order refers to the length of the longest shift register chain in the convolutional encoder. The default is 3.

- initial state specifies the initial parent state for the decode operation. When **reset?** is set to TRUE (default), the trellis structure is set to this state, thereby initializing the Viterbi decoding operation. On the first call to this VI, and thereafter when **reset?** is set to FALSE, the survivor state from the previous iteration is used to continue performing Viterbi decoding and this parameter is ignored. The default is 0.
- **traceback depth (D)** specifies the number of trellis stages used in the Viterbi decoding process. The default is 15.
- **reset?** specifies whether the decoder state is reset on each call to this VI. By default, the VI resets on the first call. When **reset?** is set to TRUE, the VI clears any buffered bits from the previous iterations, and the Viterbi hard decision decoding algorithm is initialized to start decoding from the specified **initial state**. In this case, the relationship of the length of the output decoded bit stream, *L*_{out}, to the input encoded bit stream length, *L*_{in}, is described by the following equation:

 $L_{out} = k \times [floor(L_{in}/n) - D]$

where *k/n* specifies the code rate, and *D* specifies the decoder traceback depth in symbols. When **reset?** is set to FALSE, the VI continues performing convolutional decoding from the previous iteration. In such a case, the length of the output decoded bit stream is given by the following equation:

 $L_{out} = k \times \text{floor}(L_{in}/n).$

Tip If the length of the encoded bit stream is L_{in} , and **reset?** is set to TRUE, the Viterbi decoding algorithm (for a rate k/n code) returns a total of $L_{out}=k\times[floor(L_{in}/n)-D]$ decoded bits in a single iteration, implying that a total of $k\times D$ message bits are buffered inside the VI. To recover the entire message of length $k\times floor[L_{in}/n]$ in a single call to the MT Convolutional Decoder VI with **reset?** set to TRUE, choose one of the following options:

- Append max(*D*,*K*–1)×*k* zeros to the end of the message prior to performing convolutional encoding. Here *k* represents the message word length.
- Append D×n zeros to the end of the encoded bit stream prior to calling the MT Convolutional Decoder VI. Here D is the traceback depth employed in the convolutional decoder, and n is the code word length.

The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output bit stream returns the decoded code word. Wire this parameter to the <u>MT Calculate BER</u> VI to perform bit error rate measurements.
- **final state** returns the survivor state resulting from the Viterbi hard decision decoding operation after decoding the entire **input bit stream**.
- error out passes error or warning information out of a VI to be used

by other VIs. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Convolutional Decoder (Viterbi Hard Decision, Generator Matrix)

This polymorphic instance performs Viterbi hard decision decoding on a convolutionally encoded bit stream using a custom generator matrix you specify.



input bit stream specifies the bit sequence representing the code word to decode. You can use the <u>MT Convolutional Encode</u> VI to generate the encoded bit stream. The default is empty.

generator matrix (n x k) specifies the generator connection polynomial matrix used for setting the convolutional feedforward decoder connections in octal format. The convolutional decoder is modeled as a linear feedforward shift register arrangement consisting of *k* rows with *K*-1 shift registers per row, where *k* denotes the data word length and *K* denotes the constraint length. If $a_{ij} \{0 \le i \le n-1, 0 \le j \le k-1\}$ denotes a particular element in the generator matrix, the row index *i* corresponds to the convolutional decoder output *y*_i that is affected by this element, while the column index *j* corresponds to the *j*th row in the *k* row shift register arrangement. Thus a_{ij} specifies how the *K* bits in the *j*th row of the feedforward shift register matrix affects the *i*th output of the

5

convolutional decoder. The default is $\lfloor 7 \rfloor$

For a rate 2/3 convolutional decoder, the generator matrix is specified as

 $\begin{bmatrix} 100100 & 011000 \\ 011100 & 101000 \\ 110000 & 010000 \end{bmatrix} = \begin{bmatrix} 44 & 30 \\ 34 & 50 \\ 60 & 20 \end{bmatrix}$

Notice that the preceding matrix represents the elements in octal format. Zeros are padded at the end of the corresponding code generator sequences such that the total length of the sequences is

a multiple of three.

P

Tip Ensure that the convolutional encoder and decoder receive the same generator matrices.

The following diagram depicts the rate 2/3 convolutional decoder corresponding to the generator matrix shown previously, with a constraint length of 4. In this diagram, **D** represents a shift register or memory element.



Here, y_i^{j} , $0 \le j \le n-1$ denotes the *j*th output of the convolutional decoder in the *i*th decoding instance.

- **constraint length (K)** specifies the maximum number of encoded bits that can be affected by a single input bit. This value represents (1 + maximal memory order), where maximal memory order refers to the length of the longest shift register chain in the convolutional encoder. The default is 3.
- initial state specifies the initial parent state for the decode operation. When **reset?** is set to TRUE (default), the trellis structure is set to this state, thereby initializing the Viterbi decoding operation. On the first call to this VI, and thereafter when **reset?** is set to FALSE, the survivor state from the previous iteration is used to continue performing Viterbi decoding and this parameter is ignored. The default is 0.
- **traceback depth (D)** specifies the number of trellis stages used in the Viterbi decoding process. The default is 15.
- reset? specifies whether the decoder state is reset on each call to

this VI. By default, the VI resets on the first call. When **reset?** is set to TRUE, the VI clears any buffered bits from the previous iterations, and the Viterbi hard decision decoding algorithm is initialized to start decoding from the specified **initial state**. In this case, the relationship of the length of the output decoded bit stream, L_{out} , to the input encoded bit stream length, L_{in} , is described by the following equation:

 $L_{out} = k \times [floor(L_{in}/n) - D]$

where *k/n* specifies the code rate, and *D* specifies the decoder traceback depth in symbols. When **reset?** is set to FALSE, the VI continues performing convolutional decoding from the previous iteration. In such a case, the length of the output decoded bit stream is given by the following equation:

 $L_{out} = k \times \text{floor}(L_{in}/n).$

- **Tip** If the length of the encoded bit stream is L_{in} , and **reset?** is set to TRUE, the Viterbi decoding algorithm (for a rate k/n code) returns a total of $L_{out}=k\times[floor(L_{in}/n)-D]$ decoded bits in a single iteration, implying that a total of $k\times D$ message bits are buffered inside the VI. To recover the entire message of length $k\times floor[L_{in}/n]$ in a single call to the MT Convolutional Decoder VI with **reset?** set to TRUE, choose one of the following options:
 - Append max(*D*,*K*–1)×*k* zeros to the end of the message prior to performing convolutional encoding. Here *k* represents the message word length.
 - Append *D*×*n* zeros to the end of the encoded bit stream prior to calling the MT Convolutional Decoder VI. Here *D* is the traceback depth employed in the convolutional decoder, and *n* is the code word length.

The default is TRUE.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output bit stream returns the decoded code word. Wire this parameter to the <u>MT Calculate BER</u> VI to perform bit error rate measurements.
- **final state** returns the survivor state resulting from the Viterbi hard decision decoding operation after decoding the entire **input bit stream**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - source describes the origin of the error or warning. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

MT Convolutional Decoder (Viterbi Soft Decision, Rate)

This polymorphic instance decodes a convolutionally encoded bit stream, using a specified code rate and Viterbi soft decision decoding. The code rate is equal to the ratio of the data word length to the code word length.

The following table shows the available code rates and constraint lengths supported by this polymorphic instance. If you have a particular code rate/constraint length that is not available in the table, use a generator matrix instance of the VI to specify custom values.

Code Rate	Supported Constraint Lengths
1/2	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17
1/3	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14
1/4	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14
2/3	2, 3, 4, 5, 6
3/4	2, 3, 4



input integers specifies integers representing bit metrics for quantized demodulated BPSK symbol data. These demodulated BPSK symbols are quantized to n_{soft} bits of precision, such that an integer value of 0 represents a strong logic 0 and a value of 2^n_{soft} represents a strong logic 1. The default is empty.

Note To generate the integer-valued quantized BPSK demodulated data from the Modulation Toolkit, decimate and quantize the oversampled data at the output of the <u>PSK</u> demodulator/matched filter.

For higher-order M-PSK/QAM schemes, convert the I/Q symbol values into corresponding quantized bit metrics using a suitable algorithm/heuristic.

rate (k/n) specifies the convolutional code rate as a ratio k/n, where

k is the input data word length and n is the output code word length. Use a generator matrix instance of this VI to specify a different rate. The default is 1/2.

1/2 (0)	The convolutional code rate is 1/2.
1/3 (1)	The convolutional code rate is 1/3.
1/4 (2)	The convolutional code rate is 1/4.
2/3 (3)	The convolutional code rate is 2/3.
3/4 (4)	The convolutional code rate is 3/4.

- **constraint length (K)** specifies the maximum number of encoded bits that can be affected by a single input bit. This value represents (1 + maximal memory order), where maximal memory order refers to the length of the longest shift register chain in the convolutional encoder. The default is 3.
- initial state specifies the initial parent state for the decode operation. When reset? is set to TRUE (default), the trellis structure is set to this state, thereby initializing the Viterbi decoding operation. On the first call to this VI, and thereafter when reset? is set to FALSE, the survivor state from the previous iteration is used to continue performing Viterbi decoding and this parameter is ignored. The default is 0.
- **soft decision bits** specifies the number of bits used for quantizing the BPSK modulated symbols to integers. The default is 1.
- **traceback depth (D)** specifies the number of trellis stages used in the Viterbi decoding process. The default is 15.
- **reset?** specifies whether the decoder state is reset on each call to this VI. By default, the VI resets on the first call. When **reset?** is set to TRUE, the VI clears any buffered bits from the previous iterations, and the Viterbi hard decision decoding algorithm is initialized to start decoding from the specified **initial state**. In this case, the relationship of the length of the output decoded bit stream, *L*_{out}, to the input encoded bit stream length, *L*_{in}, is described by the following equation:

 $L_{out} = k \times [floor(L_{in}/n) - D]$

where k/n specifies the code rate, and D specifies the decoder

traceback depth in symbols. When **reset?** is set to FALSE, the VI continues performing convolutional decoding from the previous iteration. In such a case, the length of the output decoded bit stream is given by the following equation:

 $L_{out} = k \times \text{floor}(L_{in}/n).$

- **Tip** If the length of the encoded bit stream is L_{in} , and **reset?** is set to TRUE, the Viterbi decoding algorithm (for a rate k/n code) returns a total of $L_{out}=k\times[floor(L_{in}/n)-D]$ decoded bits in a single iteration, implying that a total of $k\times D$ message bits are buffered inside the VI. To recover the entire message of length $k\times floor[L_{in}/n]$ in a single call to the MT Convolutional Decoder VI with **reset?** set to TRUE, choose one of the following options:
 - Append max(D,K-1)×k zeros to the end of the message prior to performing convolutional encoding. Here k represents the message word length.
 - Append *D×n* zeros to the end of the encoded bit stream prior to calling the MT Convolutional Decoder VI. Here *D* is the traceback depth employed in the convolutional decoder, and *n* is the code word length.

The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about

the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output bit stream returns the decoded code word. Wire this parameter to the <u>MT Calculate BER</u> VI to perform bit error rate measurements.
- **final state** returns the final state for the k(K-1) shift registers as the right-aligned (least significant) k(K-1) bits, where K is the constraint length, and k is the data word length in bits.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Convolutional Decoder (Viterbi Soft Decision, Generator Matrix)

This polymorphic instance performs Viterbi soft decision decoding on a convolutionally encoded bit stream using a custom generator matrix you specify.



input integers specifies integers representing bit metrics for quantized demodulated BPSK symbol data. These demodulated BPSK symbols are quantized to n_{soft} bits of precision, such that an

integer value of 0 represents a strong logic 0 and a value of 2^n soft represents a strong logic 1. The default is empty.

 $\overline{\mathbb{N}}$

Note To generate the integer-valued quantized BPSK demodulated data from the Modulation Toolkit, decimate and quantize the oversampled data at the output of the <u>PSK</u> demodulator/matched filter.

For higher-order M-PSK/QAM schemes, convert the I/Q symbol values into corresponding quantized bit metrics using a suitable algorithm/heuristic.

generator matrix (n x k) specifies the generator connection polynomial matrix used for setting the convolutional feedforward decoder connections in octal format. The convolutional decoder is modeled as a linear feedforward shift register arrangement consisting of *k* rows with *K*-1 shift registers per row, where *k* denotes the data word length and *K* denotes the constraint length. If $a_{ij} \{0 \le i \le n-1, 0 \le j \le k-1\}$ denotes a particular element in the generator matrix, the row index *i* corresponds to the convolutional decoder output *y*_i that is affected by this element, while the column index *j* corresponds to the *j*th row in the *k* row shift register arrangement. Thus a_{ij} specifies how the *K* bits in the *j*th row of the

feedforward shift register matrix affects the *i*th output of the

convolutional decoder. The default is $\lfloor 7 \rfloor$

For a rate 2/3 convolutional decoder, the generator matrix is specified as

 $\begin{bmatrix} 100100 & 011000 \\ 011100 & 101000 \\ 110000 & 010000 \end{bmatrix} = \begin{bmatrix} 44 & 30 \\ 34 & 50 \\ 60 & 20 \end{bmatrix}$

P

Notice that the preceding matrix represents the elements in octal format. Zeros are padded at the end of the corresponding code generator sequences such that the total length of the sequences is a multiple of three.

Tip Ensure that the convolutional encoder and decoder receive the same generator matrices.

The following diagram depicts the rate 2/3 convolutional decoder corresponding to the generator matrix shown previously, with a constraint length of 4. In this diagram, **D** represents a shift register or memory element.



Here, y_i^{j} , $0 \le j \le n-1$ denotes the *j*th output of the convolutional decoder in the *i*th decoding instance.

constraint length (K) specifies the maximum number of encoded bits that can be affected by a single input bit. This value represents (1 + maximal memory order), where maximal memory order refers to the length of the longest shift register chain in the convolutional encoder. The default is 3.

- initial state specifies the initial parent state for the decode operation. When **reset?** is set to TRUE (default), the trellis structure is set to this state, thereby initializing the Viterbi decoding operation. On the first call to this VI, and thereafter when **reset?** is set to FALSE, the survivor state from the previous iteration is used to continue performing Viterbi decoding and this parameter is ignored. The default is 0.
- **soft decision bits** specifies the number of bits used for quantizing the BPSK modulated symbols to integers. The default is 1.
- **traceback depth (D)** specifies the number of trellis stages used in the Viterbi decoding process. The default is 15.
- **reset?** specifies whether the decoder state is reset on each call to this VI. By default, the VI resets on the first call. When **reset?** is set to TRUE, the VI clears any buffered bits from the previous iterations, and the Viterbi hard decision decoding algorithm is initialized to start decoding from the specified **initial state**. In this case, the relationship of the length of the output decoded bit stream, *L*_{out}, to the input encoded bit stream length, *L*_{in}, is described by the following equation:

 $L_{out} = k \times [floor(L_{in}/n) - D]$

where *k/n* specifies the code rate, and *D* specifies the decoder traceback depth in symbols. When **reset?** is set to FALSE, the VI continues performing convolutional decoding from the previous iteration. In such a case, the length of the output decoded bit stream is given by the following equation:

 $L_{out} = k \times \text{floor}(L_{in}/n).$

Tip If the length of the encoded bit stream is L_{in} , and reset? is set to TRUE, the Viterbi decoding algorithm (for a rate k/n code) returns a total of $L_{out}=k\times[floor(L_{in}/n)-D]$ decoded bits in a single iteration, implying that a total of $k\times D$ message bits are buffered inside the VI. To recover the entire message of length $k\times floor[L_{in}/n]$ in a single call to the MT Convolutional Decoder VI with **reset?** set to TRUE, choose one of the following options:

- Append max(*D*,*K*–1)×*k* zeros to the end of the message prior to performing convolutional encoding. Here *k* represents the message word length.
- Append *D*×*n* zeros to the end of the encoded bit stream prior to calling the MT Convolutional Decoder VI. Here *D* is the traceback depth employed in the convolutional decoder, and *n* is the code word length.

The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output bit stream returns the decoded code word. Wire this parameter to the <u>MT Calculate BER</u> VI to perform bit error rate measurements.
- **final state** returns the final state for the k(K-1) shift registers as the right-aligned (least significant) k(K-1) bits, where K is the constraint length, and k is the data word length in bits.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and

select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Convolutional Decoder (Viterbi UnQuantized, Rate)

This polymorphic instance decodes a convolutionally encoded bit stream, using a specified code rate and unquantized Viterbi decoding. The code rate is equal to the ratio of the data word length to the code word length.

The following table shows the available code rates and constraint lengths supported by this polymorphic instance. If you have a particular code rate/constraint length that is not available in the table, use a generator matrix instance of the VI to specify custom values.

Code Rate	Supported Constraint Lengths
1/2	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17
1/3	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14
1/4	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14
2/3	2, 3, 4, 5, 6
3/4	2, 3, 4



input symbols specifies the unquantized BPSK demodulated soft symbol data. The default is empty.



Note To generate the unquantized BPSK demodulated data from the Modulation Toolkit, decimate the oversampled data at the output of the PSK demodulator/matched filter.

For higher order M-PSK/QAM schemes, convert the I/Q symbol values into corresponding bit metrics using a suitable algorithm/heuristic.

rate (k/n) specifies the convolutional code rate as a ratio *k/n*, where *k* is the input data word length and *n* is the output code word length. Use a generator matrix instance of this VI to specify a different rate. The default is 1/2.

1/2 (0) The convolutional code rate is 1/2.

1/3 (1)	The convolutional code rate is 1/3.
1/4 (2)	The convolutional code rate is 1/4.
2/3 (3)	The convolutional code rate is 2/3.
3/4 (4)	The convolutional code rate is 3/4.

- **constraint length (K)** specifies the maximum number of encoded bits that can be affected by a single input bit. This value represents (1 + maximal memory order), where maximal memory order refers to the length of the longest shift register chain in the convolutional encoder. The default is 3.
- initial state specifies the initial parent state for the decode operation. When reset? is set to TRUE (default), the trellis structure is set to this state, thereby initializing the Viterbi decoding operation. On the first call to this VI, and thereafter when reset? is set to FALSE, the survivor state from the previous iteration is used to continue performing Viterbi decoding and this parameter is ignored. The default is 0.
- **traceback depth (D)** specifies the number of trellis stages used in the Viterbi decoding process. The default is 15.
- **TEN reset?** specifies whether the decoder state is reset on each call to this VI. By default, the VI resets on the first call. When **reset?** is set to TRUE, the VI clears any buffered bits from the previous iterations, and the Viterbi hard decision decoding algorithm is initialized to start decoding from the specified **initial state**. In this case, the relationship of the length of the output decoded bit stream, *L* out, to the input encoded bit stream length, *L* in, is described by the following equation:

 $L_{out} = k \times [floor(L_{in}/n) - D]$

where *k/n* specifies the code rate, and *D* specifies the decoder traceback depth in symbols. When **reset?** is set to FALSE, the VI continues performing convolutional decoding from the previous iteration. In such a case, the length of the output decoded bit stream is given by the following equation:

 $L_{out} = k \times \text{floor}(L_{in}/n).$

Tip If the length of the encoded bit stream is L_{in} , and

P

reset? is set to TRUE, the Viterbi decoding algorithm (for a rate k/n code) returns a total of $L_{out}=k\times[floor(L_{in}/n)-D]$ decoded bits in a single iteration, implying that a total of $k\times D$ message bits are buffered inside the VI. To recover the entire message of length $k\times floor[L_{in}/n]$ in a single call to the MT Convolutional Decoder VI with **reset?** set to TRUE, choose one of the following options:

- Append max(*D*,*K*–1)×*k* zeros to the end of the message prior to performing convolutional encoding. Here *k* represents the message word length.
- Append D×n zeros to the end of the encoded bit stream prior to calling the MT Convolutional Decoder VI. Here D is the traceback depth employed in the convolutional decoder, and n is the code word length.

The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the decoded code word. Wire this

parameter to the <u>MT Calculate BER</u> VI to perform bit error rate measurements.

- **final state** returns the survivor state resulting from the Viterbi hard decision decoding operation after decoding the entire **input bit stream**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Convolutional Decoder (Viterbi UnQuantized, Generator Matrix)

This polymorphic instance performs Viterbi unquantized decoding on a convolutionally encoded bit stream using a custom generator matrix you specify.



input symbols specifies the unquantized BPSK demodulated soft symbol data. The default is empty.

 $\overline{\mathbb{N}}$

Note To generate the unquantized BPSK demodulated data from the Modulation Toolkit, decimate the oversampled data at the output of the PSK demodulator/matched filter.

For higher order M-PSK/QAM schemes, convert the I/Q symbol values into corresponding bit metrics using a suitable algorithm/heuristic.

generator matrix (n x k) specifies the generator connection polynomial matrix used for setting the convolutional feedforward decoder connections in octal format. The convolutional decoder is modeled as a linear feedforward shift register arrangement consisting of *k* rows with *K*-1 shift registers per row, where *k* denotes the data word length and *K* denotes the constraint length. If $a_{ij} \{0 \le i \le n-1, 0 \le j \le k-1\}$ denotes a particular element in the generator matrix, the row index *i* corresponds to the convolutional decoder output *y*_i that is affected by this element, while the column index *j* corresponds to the *j*th row in the *k* row shift register arrangement. Thus a_{ij} specifies how the *K* bits in the *j*th row of the feedforward shift register matrix affects the *i*th output of the

5

convolutional decoder. The default is $\lfloor 7 \rfloor$

For a rate 2/3 convolutional decoder, the generator matrix is specified as

```
\begin{bmatrix} 100100 & 011000 \\ 011100 & 101000 \\ 110000 & 010000 \end{bmatrix} = \begin{bmatrix} 44 & 30 \\ 34 & 50 \\ 60 & 20 \end{bmatrix}
```

Notice that the preceding matrix represents the elements in octal format. Zeros are padded at the end of the corresponding code generator sequences such that the total length of the sequences is a multiple of three.

Tip Ensure that the convolutional encoder and decoder receive the same generator matrices.

The following diagram depicts the rate 2/3 convolutional decoder corresponding to the generator matrix shown previously, with a constraint length of 4. In this diagram, **D** represents a shift register or memory element.



Here, y_i^{j} , $0 \le j \le n-1$ denotes the *j*th output of the convolutional decoder in the *i*th decoding instance.

- **constraint length (K)** specifies the maximum number of encoded bits that can be affected by a single input bit. This value represents (1 + maximal memory order), where maximal memory order refers to the length of the longest shift register chain in the convolutional encoder. The default is 3.
- initial state specifies the initial parent state for the decode operation. When **reset?** is set to TRUE (default), the trellis structure is set to this state, thereby initializing the Viterbi decoding operation. On the first call to this VI, and thereafter when **reset?** is

set to FALSE, the survivor state from the previous iteration is used to continue performing Viterbi decoding and this parameter is ignored. The default is 0.

- **traceback depth (D)** specifies the number of trellis stages used in the Viterbi decoding process. The default is 15.
- **TEN reset?** specifies whether the decoder state is reset on each call to this VI. By default, the VI resets on the first call. When **reset?** is set to TRUE, the VI clears any buffered bits from the previous iterations, and the Viterbi hard decision decoding algorithm is initialized to start decoding from the specified **initial state**. In this case, the relationship of the length of the output decoded bit stream, *L* out, to the input encoded bit stream length, *L* in, is described by the following equation:

 $L_{out} = k \times [floor(L_{in}/n) - D]$

where *k/n* specifies the code rate, and *D* specifies the decoder traceback depth in symbols. When **reset?** is set to FALSE, the VI continues performing convolutional decoding from the previous iteration. In such a case, the length of the output decoded bit stream is given by the following equation:

 $L_{out} = k \times \text{floor}(L_{in}/n).$

- **Tip** If the length of the encoded bit stream is L_{in} , and **reset?** is set to TRUE, the Viterbi decoding algorithm (for a rate k/n code) returns a total of $L_{out}=k\times[floor(L_{in}/n)-D]$ decoded bits in a single iteration, implying that a total of $k\times D$ message bits are buffered inside the VI. To recover the entire message of length $k\times floor[L_{in}/n]$ in a single call to the MT Convolutional Decoder VI with **reset?** set to TRUE, choose one of the following options:
 - Append max(*D*,*K*–1)×*k* zeros to the end of the message prior to performing convolutional encoding. Here *k* represents the message word length.
 - Append D×n zeros to the end of the encoded bit stream prior to calling the MT Convolutional Decoder VI. Here D is the traceback depth employed in the convolutional decoder, and n is the code word length.

The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output bit stream returns the decoded code word. Wire this parameter to the <u>MT Calculate BER</u> VI to perform bit error rate measurements.
- **final state** returns the survivor state resulting from the Viterbi hard decision decoding operation after decoding the entire **input bit stream**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The convolutional decoder receives the code words as an array of values representing Boolean bits. The decoder applies the Viterbi hard decision decoding algorithm to obtain the maximum likelihood transmitted data sequence.

The length *N* encoded input bit stream is first converted into a stream of encoded symbols of the length described by the following formula:

floor[length(input bit stream)/n]

where n is the code word length. Any bits truncated by the floor operation are buffered so that the VI can operate on a continuous stream of input bits.

The Viterbi decoder processes *D* trellis stages before determining the maximum likelihood survivor path, where *D* represents the decoder traceback depth. Typical *D* values equal $5 \times K$ stages of the trellis, where *K* denotes the constraint length of the convolutional encoder. The Viterbi decoding is implemented using the register-exchange algorithm. For reduced computational overhead in determining the path and branch metrics, the VI uses $2^{(K-2)k}$ trellis elements of size 2^k elements during intermediate calculations, where *K* is the constraint length and *k* is the data word length in bits. The following diagram represents binary soft symbol decisions from a demodulator. A sample close to -1 represents a logic 1 while a sample closer to 1 represents a logic 0.

1

-1

- Viterbi Hard Decision Decoding: In these instances, the input to the convolutional decoder corresponds to binary hard-decision data generated by the demodulator. For example, you can input the output bit stream from the MT Demodulate PSK VI or the MT Map Symbols to Bits VI to the Viterbi hard-decision decoder.
- **Unquantized Viterbi Decoding**: In these instances, the input to the convolutional decoder is represented by the I/Q symbols generated by the demodulator. These instances assume that the binary data is mapped to symbols such that a logic 0 is mapped to the I/Q location 1+0*i*, and a logical 1 is mapped to the I/Q symbol point -1+ 0*i* (corresponding to the symbol map [1+0*i* –
1+0*i*]). You can obtain the soft symbol data in the Modulation Toolkit by decimating the oversampled complex data, which is generated by the <u>MT Demodulate PSK</u> VI or the <u>MT Matched</u> <u>Filter VI</u>, to symbols using the <u>MT Decimate Oversampled</u> <u>Waveform VI</u>. These instances yield the best overall performance from the different decoder operation choices.

- Viterbi Soft-Decision Decoding: In these instances, the decoder accepts unsigned integers. Additionally, it requires you to specify the number of soft-decision bits, n_{soft} , before decoding. These soft-symbol decisions are obtained by quantizing the soft binary symbol decisions to the specified bits of precision from the demodulator, as shown in the following table. These unsigned integers range from 0 to 2^n_{soft} .
 - **Note** An eight-level quantization ($n_{soft} = 3$) yields less than a 0.25 dB reduction in coding gain relative to the unquantized Viterbi decoding. When $n_{soft} = 1$, the Viterbi soft-decision decoding algorithm reverts to performing hard-decision decoding.

The following table describes the nature of inputs fed into the Viterbi Soft-Decision decoder. This table assumes that BPSK symbol decisions from the demodulator are quantized to 3 bits of precision.

Soft Decision Input	Logic Value
0	Strong logic 0
1	Highly likely to be 0
2	Moderate 0
3	Uncertain (0 or 1)
4	Uncertain (0 or 1)
5	Moderate 1
6	Highly likely to be 1
7	Strong logic 1

Generator matrices corresponding to code rates and constraint lengths obtained from Wicker, Stephen B. *Error Control Systems*

for Digital Communications and Storage. Englewood Cliffs: Prentice-Hall, 1995.

MT Depuncture Data Stream VI

Inserts an **erasure value** inside the punctured **input data stream**. The **erasure value** is equivalent to inserting a null symbol in the position of the **input data stream** that was previously punctured using the <u>MT</u> <u>Puncture Data Stream</u> VI.

Note The VI accepts the double data type as input, as the input data stream may correspond to the real BPSK symbols at the demodulator. The erasure value symbol inserted depends on the decoding scheme employed following the depuncturing operation.

<u>Details</u>



- **input data stream** specifies the data stream with a rate to be increased by the process of depuncturing. This data stream is usually obtained from the output of a block/convolutional encoder. The default is empty.
- **puncture pattern** specifies a binary-valued matrix that dictates which elements in the **input data stream** are omitted with respect to the **output data stream**. The default is 1.
 - Note The puncture pattern matrix is chronologically ordered by columns with respect to the **input data stream**. If the dimensions of the **puncture pattern** matrix are *m×n*, the *m* elements of the first column correspond to the first *m* elements of the **input data stream**, and so on.
- erasure value specifies the symbol value to insert to the punctured input data stream before decoding. The erasure value is equivalent to inserting a null symbol in the position of the input data stream that was previously punctured using the <u>MT Puncture</u> <u>Data Stream</u> VI.
 - Note This VI accepts double data, because the input data stream can correspond to the real BPSK symbols at the demodulator. The value of the erasure symbol inserted

depends on the decoding scheme employed following the depuncturing operation. Using a soft-decision Viterbi decoder with n_{soft} bits used to quantize every BPSK modulated symbol, the integer value 0 denoting a strong logic 0, and 2^n soft-1 denoting a strong logic 1, the **erasure value** should be $(2^n \text{ soft}-1)/2$. On the other hand, if the decoder is an unquantized Viterbi decoder, which accepts real-valued BPSK data symbols lying in [-1, 1], the **erasure value** should be 0.

- **reset?** specifies whether the VI uses the stored state information from its previous iteration while puncturing the **input data stream** in the current iteration. The default is TRUE.
 - Note When the length (*L*) of the input data stream is not an integer multiple of the product *P*×*Q* of the *P*×*Q* puncture pattern matrix, the last *L* modulo (*P*×*Q*) bits are buffered inside the VI. If reset? is set to FALSE, these buffered elements are prepended to the beginning of the input data stream during the next iteration, prior to the puncturing operation.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click

the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **output data stream** returns the output data stream, which consists of all elements in the **input data stream** except those that were deleted during the puncturing operation.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

If the decoder is a soft-decision Viterbi decoder with n_{soft} bits used to quantize every BPSK modulated symbol, an integer value 0 denoting a strong logic 0, and an integer value 2^n_{soft} denoting a strong logic 1, set **erasure value** to 2^n_{soft} –1/1. However, if the decoder is an unquantized Viterbi decoder, which accepts real-valued BPSK data symbols lying in [-1, 1], set **erasure value** to 0.

MT Despread Symbols VI

Performs the <u>direct sequence spread spectrum</u> (DSSS) despreading operation on a spread sequence of data bits.



- **input chip stream** specifies the sequence of data chips to be despread.
- **spreading code** specifies the sequence of bits that serve as the spreading code.
- **sync confidence threshold** specifies the minimum correlation required for **input chip stream** to be considered synchronized with the **spreading code**. This parameter is only applicable when **reset?** is set to TRUE. Valid values are between 0.0 (no correlation required) and 1.0 (perfect correlation required).
- sync search depth specifies the desired number of bits (not chips) over which to search for a sync confidence metric greater than the specified sync confidence threshold. This parameter is only applicable when reset? is set to TRUE.
- reset? specifies whether the VI synchronizes despreading at each call using the sync search depth (bits) and sync confidence threshold parameters. reset? must be set to TRUE whenever you want to resynchronize. When reset? is set to FALSE, the sync search depth and sync confidence threshold parameters are ignored and despreading continues from the previous iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the despread series of bits.

 $\overline{\mathbb{N}}$

Note output bit stream has $(1/L \times input chip stream)$ number of elements, where *L* is the length of the **spreading** code sequence.

- **sync found index** returns the synchronization index of the synchronization sequence found within the **input chip stream**. This parameter returns –1 if no sync was found.
- **Sync confidence metric** returns the measured confidence metric when **input chip stream** is aligned using the **sync found index**. Values fall between 0.0 (0% measured confidence) and 1.0 (100% measured confidence).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Golay Decoder VI

Decodes an <u>Golay-encoded</u> bit stream. The decoder provides the two triple-correcting Golay codes: the Golay (23,12,3) code and the extended Golay (24,12,3) code. This VI uses the Arithmetic decoding algorithm for decoding the Golay (23,12,3) encoded bit stream and the Kasami error-trapping decoding algorithm for decoding the extended Golay (24,12,3) encoded bit stream.

Details



- **input bit stream** specifies the bit sequence representing the Golayencoded bits to decode. You can use the <u>MT Golay Encoder</u> VI to generate the encoded bit stream.
 - **Tip** To ensure that the **output bit stream** returned by the Golay decoder corresponds to the complete message, specify an **input bit stream** with a length that is an integer multiple of 23 (for the Golay (23,12,3) code) or 24 (for the Golay (24,12,3) code). If the length of the **input bit stream** is not an integer multiple of 23 or 24, the last {length(**input bit stream**) *modulo* 23} bits are buffered in the decoder {length(**input bit stream**) *modulo* 24} bits (for Golay (24,12,3) code). These bits are added to the beginning of the incoming bit stream during the beginning of the next iteration if **reset?** is set to FALSE.
- Golay code type (n,k,t) specifies the values of *n*, *k*, and *t* that define the code word length, data word length, and error correcting capacity, respectively, in bits.

Golay (23,12,3) (0)	The (n,k,t) values are (23, 12, 3). This is the default value.
Golay (24,12,3) (1)	The (n,k,t) values are (24, 12, 3).

reset? specifies whether to clear the internal state of the decoder. By default, the VI resets on first call. When **reset?** is set to TRUE, the VI clears any buffered bits from previous iteration. When **reset?** is set to FALSE, the VI continues Golay decoding from the previous iteration. In this case, any buffered bits from the previous iteration are added to the beginning of the **input bit stream** prior to performing Golay decoding. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **[18]** output bit stream returns the decoded bit stream.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain**

Warning from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

For further details, refer to

Wicker, Stephen B. *Error Control Systems for Digital Communications and Storage.* Englewood Cliffs: Prentice-Hall, 1995.

Lin, S. and DJ Costello, Jr. *Error Control Coding: Fundamentals and Applications.* Englewood Cliffs: Prentice-Hall, 1983.

MT Hamming Decoder VI

Decodes a Hamming-encoded bit stream.

The order-*m* Hamming codes are a special class of Bose-Chaudhari-Hocquenghem (BCH) codes in which the data word length (*k*) equals 2^m -m-1, code word length (*n*) equals 2^m -1, and the error correcting capacity (*t*) equals 1. Valid values of *m* lie in the range $2 \le m \le 16$.

input bit stream	output bit stream
hamming order	10101
error in (no error)	Hamm ¹⁰¹
reset? (T)	

input bit stream specifies the bit sequence representing the Hamming-encoded data bits to decode. Use the <u>MT Hamming</u> <u>Encoder</u> VI to generate this encoded bit stream.

Tip To ensure that the **output bit stream** data corresponds to the entire message, specify an **input bit stream** with a length that is an integer multiple of $n = 2^m - 1$ where $n = 2^m - 1$ is the length of the Hamming code word. If the length of the **input bit stream** is not an integer multiple of n, the last {length(**input bit stream**) *modulo* n} bits are buffered in the Hamming decoder. These bits are added to the beginning of the incoming bit stream during the beginning of the next iteration if **reset?** is set to FALSE.

hamming order specifies the Hamming code order. Specify an integer between 2 and 16, inclusive. Notice that the Hamming order *m* is used to specify the Hamming code with data word length equaling 2^m-m-1 and code word length equaling 2^m-1. The default is 3.

TEP reset? specifies whether to clear the internal state of the decoder. By default, the VI resets on first call. When reset? is set to TRUE, the VI initializes the Galois field structure corresponding to the Hamming code. The VI clears any buffered bits from previous iterations. When reset? is set to FALSE, the VI continues performing Hamming decoding from the previous iteration. Any buffered bits from the previous iteration are added to the beginning of the input bit stream prior to decoding. The default is TRUE. Tip When reset? is set to TRUE, the VI initializes the Galois field structure corresponding to the Hamming code. Once initialized, this same Galois field structure is used for performing the Hamming decoding provided reset? is set to FALSE. This initialization may consume significant amounts of time for code word lengths greater than 127. For best performance, perform Hamming decoding with reset? set to TRUE for the first iteration, and set to FALSE thereafter.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **[18]** output bit stream returns the decoded bit stream.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for

more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT LDPC Decoder VI

Message-passing algorithms are effective decoding strategies for lowdensity parity check (LDPC). The best algorithm known is iterative probabilistic decoding, or belief propagation. In this algorithm, at each iteration, messages are passed from message nodes to check nodes and vice versa. The message that is sent from message node (v) to the check node (c) must not take into account the message sent in the previous iteration from c to v. The same is true for messages passed from check nodes to message nodes.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT LDPC Decoder (Serial Schedule)

The LDPC serial decoding schedule is based on a serial update of symbol nodes messages. This update can be considered as shuffling of the flooding schedule. Instead of sending all messages from symbol nodes to check nodes and then all messages from check nodes to symbol nodes, as done in the flooding schedule, we go over the check nodes in some order and for each node we send all messages into the node and then all messages out from the node.



- **parity check matrix** specifies the parity check matrix used by the encoder for encoding.
- **Iikelihoods** specifies the likelihoods of the received symbols.
- **maximum number of iterations** specifies the maximum number of iterations for the iterative decoding process. The decoder stops iterating after **number of iterations** exceeds this value or if it satisfies other conditions.
- reset? (T) specifies whether to clear the internal state of the decoder. By default, the VI resets on first call.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about

the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **integrated pseudorandom data bits.** If the **PN sequence order** is *N*, the output data is periodic with period $T = 2^N - 1$. For example, if N = 7, the output sequence repeats after every T = 127 bits.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT LDPC Decoder (Flooding Schedule)

The standard message-passing schedule for decoding LDPC code is a version of the so-called flooding schedule, in which each iteration all the symbol nodes, and subsequently all the check nodes, pass new messages to their neighbors.



- **parity check matrix** specifies the parity check matrix used by the encoder for encoding.
- **likelihoods** specifies the likelihoods of the received symbols.
- **maximum number of iterations** specifies the maximum number of iterations for the iterative decoding process. The decoder stops iterating after **number of iterations** exceeds this value or if it satisfies other conditions.
- reset? (T) specifies whether to clear the internal state of the decoder. By default, the VI resets on first call.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.

- **(iii)** output bit stream returns the generated pseudorandom data bits. If the **PN sequence order** is *N*, the output data is periodic with period $T = 2^N - 1$. For example, if N = 7, the output sequence repeats after every T = 127 bits.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The decoder assumes that the parity bits are placed at the end of the code word, so it returns the first part of the code word, which are the message bits.

Belief Propagation algorithm is a sub-class of message-passing algorithms. The messages passed along the edges in this algorithm are probabilities, or beliefs. It is advantageous to work with likelihoods, or log-likelihoods instead of probabilities. For a binary random variable x, let $L(x) = \Pr[x = 0]/\Pr[x = 1]$ be the likelihood of x.

Given another random variable y, the conditional likelihood of x denoted L(x | y) is defined as Pr[x = 1|y] / Pr[x = 0|y].

The likelihood ratio is defined as L = Pr[x=1|y] / Pr[x=0|y]

For example, for a BSC channel, if the probability of error = p and the probability of no error = 1-p, then the likelihood ratio = (1-p)/p

It must be clear that the likelihood calculation depends upon the specific choice of the symbol map used in the modulation scheme. An example of likelihood calculation for a BPSK additive white Gaussian noise (AWGN) channel, if bit 0 is mapped to symbol point 1 and bit 1 is mapped to symbol point -1, is shown below.

$$P(x=0) = P_r(x=0 \mid y) = \frac{1}{1 + e^{\frac{2y}{\sigma^2}}} \qquad P(x=0) = P_r(x=0 \mid y) = \frac{1}{1 + e^{\frac{-2y}{\sigma^2}}} \qquad \frac{2y}{L = e^{\frac{-2y}{\sigma^2}}}$$

The following examples demonstrate likelihood calculations for BPSK for AWGN and BSC channels:

- MT BPSK BER vs EBN0 graph [LDPC coding]
- MT LDPC Encode Decode.vi

MT Reed Solomon Symbol Decode VI

Decodes a normal or shortened Reed-Solomon (RS) encoded integer symbol stream.

Reed-Solomon codes are specialized Bose-Chaudhari-Hocquenghem (BCH) codes in which the code elements are drawn from the Galois field $GF(2^m)$. Reed-Solomon codes are extremely powerful at burst error correction. The operations involving Reed-Solomon encoding and decoding are performed in the $GF(2^m)$, meaning that the Reed-Solomon decoder operates on code words that are symbols/groups of bits. As such, the input parameters **code word length (n)** and **data word length (k)** are specified in terms of symbols. Each symbol represents a group of $Log_2(n+1)$ bits.

Use the pull-down menu to select an instance of this VI.

Select an instance

<u>Details</u>

MT RS Symbol Decoder (normal)

This polymorphic instance decodes a Reed-Solomon encoded symbol stream in normal format. The normal Reed-Solomon (n,k,t) code consists of values of n of the form $n=2^m-1$, where valid values of m lie in the range $2 \le m \le 16$.

For example, specify n=255, k=233 to produce a Reed-Solomon (255,233,11) code. This code has an error-correcting capacity of 11 Reed-Solomon symbols.



- input symbols specifies the array of integer symbols to be decoded using the Reed-Solomon decoding scheme. Use the MT Reed-Solomon Symbol Encoder (normal) VI to generate the integer symbols required for this control. The default is empty.
 - Note The Reed-Solomon decoder is a block decoder, meaning that it operates on a block of *n* integers. To ensure that the **output symbols** stream corresponds to the entire **input symbols** array, specify an **input symbols** stream with a length *L* that is an integer multiple of *n* symbols. If the length *L* is not a multiple of *n* symbols, the last (*L* modulo *n*) symbols are buffered inside the decoder. These symbols are prepended to the incoming **input symbols** stream at the beginning of the next iteration if **reset?** is set to FALSE.
- **code word length (n), symbols** specifies the length of the Reed-Solomon code word. Each Reed-Solomon symbol represents $Log_2(n+1)$ bits. The default is 7.
- **data word length (k), symbols** specifies the length of the Reed-Solomon data word. Each Reed-Solomon symbol represents $Log_2(n+1)$ bits. The default is 3.
- **modulation session** specifies the modulation session refnum for configuring selected attributes of the Modulation Toolkit. Use this refnum to configure the advanced attributes in the <u>MT Reed</u>

Solomon Symbol Encode VI. Initialize the session with the MT Open Session VI. After configuring the modulation session refnum with the appropriate attributes, wire the modulation session refnum into the RS Symbol Encoder VI. If the RS Symbol Encoder VI is called with an uninitialized modulation session refnum, the VI returns an error. The default is empty.

- **TEP reset?** specifies whether to clear the internal state of the Reed-Solomon decoder. By default, the VI resets on first call. When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the Reed-Solomon (*n*,*k*) values. The VI clears any buffered data from previous iterations. When **reset?** is set to FALSE, the VI continues performing Reed-Solomon decoding from the previous iteration. Any buffered data from the previous iteration are prepended to the **input symbols** stream prior to decoding. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III32)** output symbols returns the Reed-Solomon decoded integer symbol stream. Wire this parameter to the <u>MT UnPack Bits</u> VI to

recover the message bits that comprise the transmitted message data.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT RS Symbol Decoder (shortened)

This polymorphic instance decodes a Reed-Solomon encoded integer symbol stream in shortened format.

Using shortened Reed-Solomon codes, the *k* and *n* values represent symbols. Each symbol has a length *m* bits, which is specified using the **bits per RS symbol** parameter. Specify a shortened RS code by specifying the shortened values of *n* and *k*, as well as the bits per symbol that correspond to the underlying Galois field arithmetic. For example, you can specify n=32, k=28, and **bits per RS symbol**=8, to generate a shortened Reed-Solomon (32,28,2) code with error-correcting capacity of 2 Reed-Solomon symbols. For this code, the underlying Galois field arithmetic is performed in GF(2^8)=256.



input symbols specifies the array of integer symbols to be decoded using the shortened Reed-Solomon decoding scheme. Use the MT Reed-Solomon Symbol Encoder (shortened) VI to generate the integer symbols required for input to this control. The default is empty.



Note The shortened Reed-Solomon encoder is a block encoder, meaning that it operates on a block of *n* integers. To ensure that the **output symbols** stream corresponds to the entire **input symbols** array, specify an **input symbols** stream with a length *L* that is an integer multiple of n symbols. If the length *L* is not a multiple of *n* symbols, the last (*L* modulo *n*) symbols are buffered inside the encoder. These symbols are prepended to the incoming **input symbols** stream at the beginning of the next iteration if **reset?** is set to FALSE.

code word length (n), symbols specifies the length of the Reed-Solomon code word. Each Reed-Solomon symbol represents $Log_2(n+1)$ bits. The default is 7.

data word length (k), symbols specifies the length of the Reed-Solomon data word. Each Reed-Solomon symbol represents $Log_2(n+1)$ bits. The default is 3.

modulation session specifies the modulation session refnum for configuring selected attributes of the Modulation Toolkit. Use this refnum to configure the advanced attributes in the MT Reed Solomon Symbol Encode VI. Initialize the session with the MT Open Session VI. After configuring the modulation session refnum with the appropriate attributes, wire the modulation session refnum into the RS Symbol Encoder VI. If the RS Symbol Encoder VI is called with an uninitialized modulation session refnum, the VI returns an error. The default is empty.

- bits per RS symbol specifies the number of bits per RS symbol. This parameter specifies the Galois field over which the encoding operation is performed as GF(2^{bits per RS symbol}). Valid values fall in the range 3–16 inclusive. For example, to specify a shortened Reed-Solomon (32,28) code in the Galois field GF(256), specify code word length (n) symbols as 32 symbols, code word length (k) symbols as 28 symbols, and bits per RS symbol as 8. The default is 3.
- **TEN reset?** specifies whether to clear the internal state of the Reed-Solomon decoder. By default, the VI resets on first call. When **reset?** is set to TRUE, the VI initializes the Galois field structure corresponding to the Reed-Solomon (*n*,*k*) values. The VI clears any buffered data from previous iterations. When **reset?** is set to FALSE, the VI continues performing Reed-Solomon decoding from the previous iteration. Any buffered data from the previous iteration are prepended to the **input symbols** stream prior to decoding. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output symbols** returns the Reed-Solomon decoded integer symbol stream. Wire this parameter to the <u>MT UnPack Bits</u> VI to recover the message bits that comprise the transmitted message data.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The values of (n,k) passed to the RS decoder are specified in terms of RS symbols, where *n* denotes the RS code word length (symbols) and *k* denotes the RS data word length (symbols). For decoding, you can choose either a normal RS code or a shortened RS code. Each normal RS-encoded symbol corresponds to $Log_2(n + 1)$ bits. Each shortened RS-encoded symbol corresponds to the number of bits specified in the **bits per RS symbol** parameter.

Code Type	Bits per RS Symbol (<i>m</i>)	Code Word Length (<i>n</i>)	Data Word Length (<i>k</i>)	Error- Correcting Capacity (<i>t</i>)
Reed- Solomon (normal)	Log ₂ (n + 1)	2 ^m - 1	2m - 1 - 2t	(n - k)/2
Reed- Solomon (shortened)	user-specified	2 ^m - 1 - L	2m - 1 - L - 2t	(n - k)/2

The following table relates RS code types to data and word lengths, where L is the number of shortening zeroes.

DeInterleaving

Owning Palette: Digital

Use the VIs on the **Deinterleaving** palette to perform depermutation, and block and convolutional deinterleaving.

Palette Object	Description
<u>MT Block</u> Deinterleave (Matrix)	Performs deinterleaving by writing the input data into an m (rows) × n (columns) matrix, column by column, and then reading the data out row by row when the matrix is full. Typically, block interleaver and deinterleaver functions are used together.
<u>MT</u> DePermute	Rearranges, or depermutes, the permuted data.
<u>MT</u> <u>Convolutional</u> <u>DeInterleave</u>	Performs deinterleaving using delays that are computed based on the delays defined for the corresponding interleaver.

MT Convolutional DeInterleave VI

Performs deinterleaving using delays that are computed based on the delays defined for the corresponding interleaver.

A convolutional deinterleaver is the inverse of a convolutional interleaver. In convolutional deinterleaving, data elements pass cyclically through a set of branches. That is, in an *N* branch convolutional deinterleaver, the element 0 goes through branch 0, element 1 goes through branch 1, element *N*–1 goes through branch *N*–1, element *N* goes through branch 0, and so on. Each branch has different delays associated with it. Hence the data sent to each deinterleaver branch is delayed by a specific amount (the amount of delay in that particular branch) before the deinterleaver returns the data.

In a convolutional interleaver, if the delay in branch number n is d_n , maximum delay is *max_delay* and the minimum delay is *min_delay*, then for the corresponding convolutional deinterleaver, the delay in branch number n is:

•

D_n=(max_delay+min_delay)-d_n.

Use the pull-down menu to select an instance of this VI.

Select an instance

<u>Details</u>

MT Convolutional DeInterleaver (Standard)

This polymorphic instance performs the deinterleaving process using a fixed number of branches and fixed unit delay.



- **data in** specifies the input data to the deinterleaver. The default is empty.
- **number of branches** specifies the number of branches of the convolutional deinterleaver. Data elements pass through the branches in a cyclic fashion. For example, in an *N* branch convolutional deinterleaver, data element 0 goes through branch 0, element 1 goes through branch 1, element *N*–1 goes through branch 0, branch *N*–1, element *N* returns through branch 0 and so on. Each branch incorporates different delays. The default is empty.
- **unit delay** specifies the unit delay value. If this value is defined as D, then the number of delays on the *i*th branch is (*i*×*D*). If the total number of branches is *N*, then *i*=0,1,..., *N*–1. The default is 1.
- **initial state** specifies the shift register values when the convolutional deinterleaver begins operation. The default is empty.
- **reset?** specifies whether to check the current input parameters. On the first call to this VI or when **reset?** is set to TRUE, the VI checks input parameters. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[132] data out returns the output of the convolutional deinterleaver.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Convolutional DeInterleaver (Generic)

This polymorphic instance performs the deinterleaving process with userspecified delays for each branch.



- **data in** specifies the input data to the deinterleaver. The default is empty.
- **interleaver delays** specifies the number of delays individually for each branch of the corresponding interleaver. The number of elements in this array determines the number of branches, and the value of each element denotes the delay for that branch. For example, if **interleaver delays** is [3,6,9,2], then it implies that branch 0 has 3 delays, branch 1 has 6 delays, and so on, for the corresponding interleaver. The default is empty.



Note You must specify a value for this parameter.

- **initial state** specifies the shift register values when the convolutional deinterleaver begins operation. The default is empty.
- **reset?** specifies whether to check the current input parameters. On the first call to this VI or when **reset?** is set to TRUE, the VI checks input parameters. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front

panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[132] data out returns the output of the convolutional deinterleaver.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
Details

The following example demonstrates convolutional interleaving/deinterleaving.

Let the **data in** be: x0, x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, x11, x12, x13, ...



D = unit delay in the path

Assume for this example that the unit delay = 1, and that the initial state shift registers are initialized with values of 0 for both the interleaver and the deinterleaver.

Interleaved Data: x0, 0, 0, 0, x4, x1, 0, 0, x8, x5, x2, 0, x12, x9,

References

- 1. Wicker, Stephen B. *Error Control Systems for Digital Communication and Storage.* Englewood Cliffs: Prentice-Hall, 1995.
- 2. Proakis, John G. and Masoud Salehi. *Communication Systems Engineering.* 2d ed. Englewood Cliffs: Prentice-Hall, 2001.
- 3. Sklar, Bernard. *Digital Communications: Fundamentals and Applications.* 2d ed. Englewood Cliffs: Prentice-Hall, 2001.

MT DePermute VI

Rearranges, or depermutes, the permuted data.

Details



- **data in** specifies input data to be depermuted so that the generated data is the same as the <u>MT Permute</u> VI input data. The default is empty.
- **permutation array** specifies an array with elements that define the location of the output data samples of the <u>MT Permute</u> VI. You can use this array to track the locations of elements in the original data input to the <u>MT Permute</u> VI. The default is empty.
- **TEP** reset? specifies how the VI handles buffered data. When reset? is set to TRUE, buffered data is cleared. Input parameters are checked on a first call to this VI. Any change in the input parameter values during subsequent iterations of the VI is reflected only if reset? is set to TRUE. In continuous operations, when the buffered data is appended in the beginning of data from next iteration, set reset? to FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **[132]** data out returns the depermuted data.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Refer to the following theoretical example for an illustration of the depermutation operation.

If the **data in** array is entered as 9,21,13,7,17, and the **permutation array** is 3,2,4,0,1, the **data out** array would be 7,13,17,9,2, as illustrated in the following figure:



Demodulation

Owning Palette: Digital

Use the VIs on the **Demodulation** palette to implement several types of digital demodulation.

Palette Object	Description
<u>MT</u> <u>Demodulate</u> <u>ASK</u>	Demodulates an ASK-modulated complex baseband waveform and returns the time-aligned oversampled complex waveform, the demodulated bit stream, and the results of offset and drift measurements. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
MT Demodulate PAM	Demodulates a pulse-amplitude-modulated (PAM) complex baseband waveform and returns the time-aligned oversampled complex waveform, the demodulated bit stream, and the results of offset and drift measurements. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
MT Demodulate QAM	Demodulates a QAM-modulated complex baseband waveform and returns the time-aligned oversampled complex waveform, the demodulated bit stream, and the results of offset and drift measurements. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
<u>MT</u> Demodulate FSK	Demodulates an FSK-modulated complex baseband waveform and returns the time-aligned demodulated waveform, the demodulated information bit stream, and measurement results obtained during demodulation. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
MT Demodulate MSK	Demodulates an MSK-modulated complex baseband waveform and returns the time-aligned oversampled complex waveform, the ideal oversampled waveform, the demodulated bit stream, and the results of offset and drift measurements. This VI attempts to remove carrier and

	phase offset by locking to the carrier signal.
<u>MT</u> Demodulate PSK	Demodulates a PSK-modulated complex baseband waveform and returns the time-aligned oversampled complex waveform, the demodulated bit stream, and measurement results obtained during demodulation. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
MT Demodulate CPM	Demodulates a CPM-modulated complex baseband waveform and returns the time-aligned demodulated waveform, the demodulated information bit stream, and measurement results obtained during demodulation. This VI attempts to remove carrier and phase offset by locking to the carrier signal.

MT Demodulate ASK VI

Demodulates an ASK-modulated complex baseband waveform and returns the time-aligned oversampled complex waveform, the demodulated bit stream, and the results of offset and drift measurements. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI assumes that the sample rate of the **input complex** waveform is exactly **samples per symbol** × the symbol rate. If this relationship does not apply to your application, use the <u>MT</u> <u>Resample (Complex Cluster)</u> VI to resample the waveform to the desired sample rate.



Note Matched filtering and/or waveform realignment performed during symbol timing recovery may lead to the apparent loss of bits. Refer to Filter Delay for more information about this effect. You can use the MT Detect ASK VI if your application requires only the demodulated bit stream output and not the **recovered complex waveform** or **measurements**.

Details



- **input complex waveform** specifies the ASK-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the acquired **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- ASK system parameters specifies parameter values defining the ASK system. Wire the ASK system parameters cluster from the

MT Generate System Parameters VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



N

Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

symbol map specifies an ordered array that maps each symbol to its desired level. The number of ASK levels in the array is 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin is 1.

matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the ASK system parameters cluster that is passed to this VI.

Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.

synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the ASK synchronization parameters cluster returned by the ASK bit array or number array instances of the MT Generate Synchronization Parameters VI to this cluster.

Note If the **synchronization parameters** cluster is not wired, the demodulator does not attempt to synchronize, and

the constellation of the demodulated waveform has a 180° carrier phase ambiguity.

- expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
- **sync sequence** specifies the mapped symbol pattern used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty (default), the signal is still demodulated, but there is a phase ambiguity in the recovered symbols.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located.
- **Sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.
- **reset?** specifies whether the VI continues demodulating using the previous iteration states. **reset?** must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE if the **input complex waveform** is contiguous with the **input complex waveform** from the previous iteration of this VI. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **recovered complex waveform** returns the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. The frequency offset and phase offset corrections are scalar values applied to the entire block.
 - **IDENTIFY and SET UP: IDENTIFY and SET UP:**
 - **DEED** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **[18]** output bit stream returns the demodulated information bit stream.
 - **Note** For ASK systems with more than 1 bit per symbol, such as 4-ASK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- measurements returns the results of measurements performed by the ASK demodulator.
 - **frequency offset** returns the measured carrier frequency offset, in hertz. The measured frequency offset is removed from the **recovered complex waveform**.
 - **IFE** frequency drift returns the measured carrier frequency drift,

in hertz. The measured frequency drift is not removed from the **recovered complex waveform**.

- **phase offset** returns the measured phase offset, in degrees. The measured phase offset is removed from the **recovered complex waveform**.
- **Sync found index** returns the symbol index within the **input complex waveform** where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Demodulate CPM VI

<u>Demodulates</u> a <u>CPM-modulated</u> complex <u>baseband</u> waveform and returns the time-aligned demodulated waveform, the demodulated information bit stream, and measurement results obtained during demodulation. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI assumes that the sample rate of the **input complex** waveform is exactly **samples per symbol** × the symbol rate. If this relationship does not apply to your application, use the <u>MT</u> <u>Resample (Complex Cluster)</u> VI to resample the waveform to the desired sample rate.



Note Matched filtering and/or waveform realignment performed during symbol timing recovery may lead to the apparent loss of bits. Refer to <u>Filter Delay</u> for more information about this effect. You can use the <u>MT Detect CPM</u> VI if your application requires only the demodulated bit stream output and not the **recovered complex waveform** or **measurements**.

Details



- **input complex waveform** specifies the CPM-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.

Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively. **CPM system parameters** specifies parameter values defining the CPM system. Wire the **CPM system parameters** cluster from the CPM (M) or CPM (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



- **Note** The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.
- **symbol map** specifies the number of samples per symbol in the modulated output complex waveform. It should be an even number.
- **matched filter coefficients** specifies an ordered array containing the desired matched filter coefficients. Wire the **matched filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **matched samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **CPM system parameters** parameter in this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to prime the filter, hence eliminating the transient.
 - **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L/2 + P/2 + 4(13 + K)]/K, where *L* is the matched filter length in taps, *P* is the pulse-shaping filter length in taps, and *K* is the number of samples per symbol. For typical values of *L* = 57, *P* = 25, and *K* = 4, this value equals 27.25 symbols. Therefore when **reset?** is set to TRUE, a total of 1,028 CPM symbols must be sent to the demodulator to obtain at least 1,000 symbols at the

output. This formula does not account for truncation due to any specified **synchronization parameters**.

- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for the sequence. Wire the **CPM synchronization parameters** cluster returned by the CPM bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.
 - Note If the CPM synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a carrier phase ambiguity.
 - expected sync location specifies the expected location of the first symbol of the synchronization sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **Sync sequence** specifies the mapped symbol pattern. Although the data type is complex, only the real portion is used. For CPM, the real portion of the mapped symbols is the <u>frequency deviation</u> of the symbol value, and the imaginary portion is 0. To prevent false synchronization, configure this pattern so that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty, the signal is still demodulated.
 - **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
 - **sync indent** specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.

pulse shaping filter coefficients specifies an ordered array containing the desired pulse-shaping coefficients. This parameter is used internally to reproduce the ideal waveform for making measurements. Wire the **pulse shaping filter coefficients** parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI to the samples per symbol parameter of the samples per symbol parameters cluster.

reset? (T) specifies an ordered array containing the desired pulseshaping coefficients. This parameter is used internally to reproduce the ideal waveform for making measurements. Wire the pulse shaping filter coefficients parameter of the MT Generate Filter <u>Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the pulse shaping samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the CPM system parameters cluster.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.

- **recovered complex waveform** returns the time-aligned and oversampled complex waveform data after frequency offset correction and phase offset correction. The frequency offset and phase offset corrections are scalar values applied to the entire block.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **besilon dt** returns the time between values in the **Y** array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.

[18] output bit stream returns the demodulated information bit stream.

- **Note** For CPM systems with more than 1 bit per symbol, such as 4-CPM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- measurements returns the results of measurements performed by the CPM demodulator.
 - **frequency offset** returns the measured carrier frequency offset, in hertz. The measured frequency offset is removed from the **recovered complex waveform**.
 - **FINEL** frequency drift returns the measured carrier frequency drift, in hertz. The measured frequency drift is not removed from the recovered complex waveform.
- **Sync found index** returns the symbol index within the **input complex waveform** where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.
- **detected complex waveform** returns the ideal oversampled waveform corresponding to the **output bit stream**. Wire this

parameter to the **detected complex waveform** parameter of CPM instance of the <u>MT Measure Quadrature Impairments</u> VI.

t0 returns the trigger (start) time of the Y array.

besilt of the time between values in the Y array.

- Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

NI recommends using some form of pulse shaping on continuous-phase CPM-modulated signals to optimize demodulator performance. Demodulator performance under these conditions can also be improved by increasing the number of samples per symbol, but you can achieve better performance by using some form of pulse shaping.

MT Demodulate FSK VI

<u>Demodulates</u> an <u>FSK-modulated</u> complex <u>baseband</u> waveform and returns the time-aligned demodulated waveform, the demodulated information bit stream, and measurement results obtained during demodulation. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI assumes that the sample rate of the **input complex** waveform is exactly **samples per symbol** × the symbol rate. If this relationship does not apply to your application, use the <u>MT</u> <u>Resample (Complex Cluster)</u> VI to resample the waveform to the desired sample rate.



Note Matched filtering and/or waveform realignment performed during symbol timing recovery may lead to the apparent loss of bits. Refer to <u>Filter Delay</u> for more information about this effect. You can use the <u>MT Detect FSK</u> VI if your application requires only the demodulated bit stream output and not the **recovered complex waveform** or **measurements**.

Details



- **input complex waveform** specifies the FSK-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.

FSK system parameters specifies parameter values defining the

FSK system. Wire the **FSK system parameters** cluster returned by the FSK (M) or FSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- Note If the symbol phase continuity element of the FSK system parameters cluster is set to discontinuous, no pulse-shaping filter can be applied, and the matched filter coefficients parameter is ignored. Refer to the FSK (M) or FSK (Map) instance of the MT Generate System Parameters VI for more information about these parameters.
- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - $\overline{\mathbb{N}}$

Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

- **symbol map** specifies an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of FSK levels in the array must be 2^N, where N is the number of bits per symbol.
- **symbol phase continuity** specifies whether the <u>phase</u> transitions between symbols are continuous.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, discontinuous phase FSK (DPFSK).

matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the FSK system parameters cluster passed to this VI.

- Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L/2 + P/2 + 4(13 + K)]/K, where *L* is the matched filter length in taps, *P* is the pulse-shaping filter length in taps, and *K* is the number of samples per symbol. For typical values of L = 57, P = 25, and K = 4, this value equals 27.25 symbols. Therefore, when **reset?** is set to TRUE, a total of 1,028 FSK symbols must be specified to the demodulator to obtain at least 1,000 symbols at the output. This formula does not account for truncation caused by any specified **synchronization parameters**.
- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for the sequence. Wire the FSK synchronization parameters cluster returned by the FSK bit array or number array instances of the MT Generate Synchronization Parameters VI to this cluster.
 - Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a carrier phase ambiguity.
 - expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This

parameter is ignored if the **expected sync location** parameter is set to -1.

- **Sync sequence** specifies the mapped symbol pattern. Although the data type is complex, only the real portion is used. For FSK, the real portion represents the frequency deviation of the symbol used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. For FSK, the real portion of the mapped symbols is the <u>frequency deviation</u> of the symbol value, and the imaginary portion is 0. If this parameter is left empty, the signal is still demodulated.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.
- **pulse shaping filter coefficients** specifies an ordered array containing the desired pulse shaping coefficients. This parameter is used internally to reproduce the ideal waveform for making measurements. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **FSK system parameters** cluster to this VI.
- **reset?** specifies whether the VI continues demodulating using the previous iteration states. **reset?** must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE if the **input complex waveform** is contiguous with the **input complex waveform** from the previous iteration of this VI. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

recovered complex waveform returns the time-aligned and oversampled complex waveform data after frequency offset correction and phase offset correction. The frequency offset and phase offset corrections are scalar values applied to the entire block.

Note The recovered complex waveform returned by the FSK demodulator is corrected for carrier phase and frequency offsets. Because FSK modulation is essentially a digital implementation of analog FM modulation, you must perform FM demodulation and matched filtering to make frequency deviation measurements or display the eye diagram of the frequency of the recovered waveform. To do this, pass the recovered complex waveform to the MT Demodulate FM VI, followed by the MT Matched Filter VI. Refer to the VI named mod_Plot FSK Eye Diagram.vi, available in the <LabVIEW >/Modulation/Digital/Support/Examples folder, for an example of how to create and eye diagram.

ID returns the trigger (start) time of the **Y** array.

DEED dt returns the time interval between data points in the Y array.

Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.

[18] output bit stream returns the demodulated information bit stream.

- **Note** For FSK systems with more than 1 bit per symbol, such as 4-FSK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- measurements returns the results of measurements performed by the FSK demodulator.
 - **frequency offset** returns the measured carrier frequency offset, in hertz. The measured frequency offset is removed from the **recovered complex waveform**.
 - **frequency drift** returns the measured carrier frequency drift, in hertz. The measured frequency drift is not removed from the **recovered complex waveform**.
 - deviation error returns the measured <u>FSK deviation error</u>, in hertz.
 - **FSK error** returns the measured <u>FSK error</u>, in hertz. This value is an average of the average rms deviation error for each FSK state.

FSK error can be measured as a percentage by normalizing the reported FSK error, in hertz, by the maximum ideal frequency deviation and multiplying by 100, as shown in the following formula:

FSK error(%) = [FSK error(Hz)/maximum ideal frequency deviation]×100

The maximum ideal frequency deviation can be obtained from the FSK **symbol map**.

sync found index returns the symbol index within the input complex waveform where the peak correlation to the sync sequence was found. If no sync sequence is specified in the synchronization parameters cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Tip NI recommends using some form of pulse shaping on continuous-phase FSK-modulated signals to optimize demodulator performance. Demodulator performance under these conditions can also be improved by increasing the number of samples per symbol, but a better improvement is realized by using some form of pulse shaping.

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Demodulate MSK VI

<u>Demodulates</u> an <u>MSK-modulated</u> complex <u>baseband</u> waveform and returns the time-aligned oversampled complex waveform, the ideal oversampled waveform, the demodulated bit stream, and the results of offset and drift measurements. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI assumes that the sample rate of the **input complex** waveform is exactly **samples per symbol** × the symbol rate. If this relationship does not apply to your application, use the <u>MT</u> <u>Resample (Complex Cluster)</u> VI to resample the waveform to the desired sample rate.



Note Matched filtering and/or waveform realignment performed during symbol timing recovery may lead to the apparent loss of bits. Refer to <u>Filter Delay</u> for more information about this effect. You can use the <u>MT Detect MSK</u> VI if your application requires only the demodulated bit stream output and not the **recovered complex waveform** or **measurements**.

Details

205



- **input complex waveform** specifies the MSK-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.

Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively. **MSK system parameters** specifies parameter values defining the MSK system. Wire the **MSK system parameters** cluster returned by the MSK instance of the <u>MT Generate System Parameters</u> VI to this cluster. Do not alter the values.

differential encoding enable specifies the status of the differential encoding.

disable (0)	Does not allow differential encoding for input symbols.
enable (1)	Allows differential encoding for input symbols. This is the default value.

samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

- matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the MSK system parameters cluster passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
 - **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L/2 + P/2 + 4(13 + K)]/K, where *L* is the matched filter length in taps, *PP* is the pulse-shaping filter length in taps, and *K* is the number of samples

per symbol. For typical values of L = 11, P = 25, and K = 4, this value equals 21.5 symbols. Therefore when **reset?** is set to TRUE, a total of 1,022 MSK symbols must be specified to the demodulator to obtain at least 1,000 symbols at the output. This formula does not account for truncation due to any specified **synchronization parameters.**

- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the MSK synchronization parameters cluster returned by the MSK bit array or number array instances of the MT Generate Synchronization Parameters VI to this cluster.
 - Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a carrier phase ambiguity.
 - expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync sequence** specifies the mapped symbol pattern used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. For MSK, the real portion of the mapped symbols is the <u>frequency deviation</u> of the symbol value, and the imaginary portion is 0. If this parameter is left empty, the signal is still demodulated.
 - **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located.
 - sync indent specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the

output bit stream consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.

- **pulse shaping filter coefficients** specifies an ordered array containing the desired pulse shaping coefficients. This parameter is used internally to remodulate the bit stream to calculate measurements. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **MSK system parameters** cluster passed to this VI.
- **reset?** specifies whether the VI continues demodulating using the previous iteration states. **reset?** must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE if the **input complex waveform** is contiguous with the **input complex waveform** from the previous iteration of this VI. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - source describes the origin of the error or warning. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

- **recovered complex waveform** returns the time-aligned and oversampled complex waveform data after frequency offset correction and phase offset correction. The frequency offset and phase offset corrections are scalar values applied to the entire block.
 - Note The recovered complex waveform returned by the MSK demodulator is corrected for carrier phase and frequency offsets. As MSK modulation is essentially a digital implementation of analog FM modulation, you must perform FM demodulation and matched filtering to make frequency deviation measurements or display the eye diagram of the frequency of the recovered waveform. To do this, pass the **recovered complex waveform** to the MT Demodulate FM VI, followed by the MT Matched Filter VI. Refer to the VI named mod_Plot FSK Eye Diagram, available in the <LabVIEW >/Modulation/Digital/Support/Examples folder, for an example of how to create an eye diagram.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **besilt of the time interval between data points in the Y array.**
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- **[18]** output bit stream returns the demodulated information bit stream.
- measurements returns the results of measurements performed by the MSK demodulator.
 - **frequency offset** returns the measured carrier frequency offset, in hertz. The measured frequency offset is removed from the **recovered complex waveform**.
 - **frequency drift** returns the measured carrier frequency drift, in hertz. The measured frequency drift is not removed from the **recovered complex waveform**.
- **IDENTIFY and IDENTIFY STATES SET UP: SYNC FOUND INDEX WITHIN THE INPUT**

complex waveform where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.

- detected complex waveform returns the ideal oversampled waveform corresponding to the output bit stream. Wire this parameter to the detected complex waveform parameter of the MSK instance of the <u>MT Measure Quadrature Impairments</u> VI.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **DEED** dt returns the time interval between data points in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Demodulate PAM VI

<u>Demodulates</u> a pulse-amplitude-<u>modulated</u> (PAM) complex <u>baseband</u> waveform and returns the time-aligned oversampled complex waveform, the demodulated bit stream, and the results of offset and drift measurements. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI assumes that the sample rate of the **input complex** waveform is exactly **samples per symbol** × the symbol rate. If this relationship does not apply to your application, use the <u>MT</u> <u>Resample (Complex Cluster)</u> VI to resample the waveform to the desired sample rate.



Note Matched filtering and/or waveform realignment performed during symbol timing recovery may lead to the apparent loss of bits. Refer to <u>Filter Delay</u> for more information about this effect. You can use the <u>MT Detect PAM</u> VI if your application requires only the demodulated bit stream output and not the **recovered complex waveform** or **measurements**.

Details



- **input complex waveform** specifies the PAM-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies time interval between data points in the acquired **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- **PAM system parameters** specifies parameter values defining the PAM system. Wire the **PAM system parameters** cluster returned

by the PAM (M) or PAM (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

symbol map specifies an ordered array that maps each symbol to its desired level. The number of PAM levels in the array is 2^N, where N is the number of bits per symbol. The vector length for the symbol(s) farthest from the origin is 1.

matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the PAM system parameters cluster that is passed to this VI.

- Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for the sequence. Wire the **PAM synchronization parameters** cluster returned by the PAM bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.



Note If the **synchronization parameters** cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a carrier
phase ambiguity.

- expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
- **sync sequence** specifies the mapped symbol pattern. Although the data type is complex, only the real portion is used. For CPM, the real portion of the mapped symbols is the <u>frequency deviation</u> of the symbol value, and the imaginary portion is 0. To prevent false synchronization, configure this pattern so that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty, the signal is still demodulated.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- sync indent specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the output bit stream consists of 10 data symbols, followed by the sync sequence, followed by the remaining data symbols.
- **reset?** specifies whether the VI continues demodulating using the previous iteration states. **reset?** must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE if the **input complex waveform** is contiguous with the **input complex waveform** from the previous iteration of this VI. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **recovered complex waveform** returns the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. The frequency offset and phase offset corrections are scalar values applied to the entire block.
 - **10** returns the time of first value in **Y** array.
 - **DEL** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.

[18] output bit stream returns the demodulated information bit stream.

- **Note** For PAM systems with more than 1 bit per symbol, such as 4-PAM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- measurements returns the results of measurements performed by the PAM demodulator.
 - **frequency offset** returns the measured carrier frequency offset, in hertz. The measured frequency offset is removed

from the recovered complex waveform.

- **FINIL** frequency drift returns the measured carrier frequency drift, in hertz. The measured frequency drift is not removed from the recovered complex waveform.
- **phase offset** returns the measured phase offset, in degrees. The measured phase offset is removed from the **recovered complex waveform**.
- **Sync found index** returns the symbol index within the **input complex waveform** where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Demodulate PSK VI

<u>Demodulates</u> a <u>PSK-modulated</u> complex <u>baseband</u> waveform and returns the time-aligned oversampled complex waveform, the demodulated bit stream, and measurement results obtained during demodulation. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI assumes that the sample rate of the **input complex** waveform is exactly **samples per symbol** × the symbol rate. If this relationship does not apply to your application, use the <u>MT</u> <u>Resample (Complex Cluster)</u> VI to resample the waveform to the desired sample rate.



Note Matched filtering and/or waveform realignment performed during symbol timing recovery may lead to the apparent loss of bits. Refer to Filter Delay for more information about this effect. You can use the MT Detect PSK VI if your application requires only the demodulated bit stream output and not the recovered complex waveform or measurements.

Details



- **input complex waveform** specifies the PSK-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies time interval between data points in the acquired **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.

PSK system parameters specifies parameter values defining the

PSK system. Wire the **PSK system parameters** cluster of the PSK (M) or PSK (Map) instance of the <u>MT Generate System Parameters</u> VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

symbol map specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.

differential PSK specifies the status of the differential PSK.

disable (0) Does not allow bit sequence encoding.

enable (1) Allows bit sequence encoding.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.	
shifted	Rotates the constellation by π/M each symbol.	
offset	Sets the modulation type to offset quadrature phase- shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from the ideal symbol timing for I. offset is currently only supported for $M = 4$.	

matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **PSK system parameters** cluster passed to this VI.

- Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L + 4(13 + K)]/K, where *L* is the matched filter length in taps and *K* is the number of samples per symbol. For typical values of *L* = 25 (4 samples per symbol, filter length of 6 symbols) and *K* = 4, this value equals 23.25 symbols. Therefore when **reset?** is set to TRUE, a total of 1,024 PSK symbols must be input to the demodulator to obtain at least 1,000 symbols at the output. This formula does not account for truncation due to any specified **synchronization parameters**.
- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the **PSK synchronization parameters** cluster returned by the PSK bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.
 - Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a 180° carrier phase ambiguity.
 - expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync sequence** specifies the mapped symbol pattern used to

synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty (default), the signal is still demodulated, but there is a phase ambiguity in the recovered symbols.

- sync location uncertainty specifies the number of symbols before or after the expected sync location where the first symbol of the sync sequence may be located.
- sync indent specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the output bit stream consists of 10 data symbols, followed by the sync sequence, followed by the remaining data symbols.

pulse shaping filter coefficients specifies an ordered array containing the desired pulse shaping coefficients. This parameter is used internally to remodulate the bit stream to calculate measurements. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the **MT Generate Filter Coefficients** VI to the samples per symbol element of the **PSK system parameters** cluster passed to this VI.

- **reset?** specifies whether the VI continues demodulating using the previous iteration states. **reset?** must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE if the **input complex waveform** is contiguous with the **input complex waveform** from the previous iteration of this VI. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain

Error or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **recovered complex waveform** returns the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. The frequency offset and phase offset corrections are scalar values applied to the entire block.
 - **t0** returns the time of first value in **Y** array.
 - **besilved between values in the Y array**.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.

[18] output bit stream returns the demodulated information bit stream.

Note For PSK systems with more than 1 bit per symbol, such as 4-PSK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...

measurements returns the results of measurements performed by the PSK demodulator.

Interpretation of the set of the

offset, in hertz. The measured frequency offset is removed from the **recovered complex waveform**.

- **frequency drift** returns the measured carrier frequency drift, in hertz. The measured frequency drift is not removed from the **recovered complex waveform**.
- phase offset returns the measured phase offset, in degrees. The measured phase offset is removed from the recovered complex waveform.

sync found index returns the symbol index within the input complex waveform where the peak correlation to the sync sequence was found. If no sync sequence is specified in the synchronization parameters cluster, this parameter returns the offset from the start of the input complex waveform to the first complete symbol.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Note Offset PSK mode is only supported for 4-PSK (OQPSK), and setting **psk type** to **offset** requires some form of band-limiting filtration. Raised cosine and root-raised cosine are the only valid filter types for OQPSK demodulation. Demodulator locking performance degrades with alpha values <0.3 in the raised cosine or root-raised cosine filter design. Set the **PSK type** parameter to offset in the PSK (M) or PSK (Map) instance of <u>MT Generate</u> <u>System Parameters</u> to access this mode.

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Demodulate QAM VI

Demodulates a QAM-modulated complex baseband waveform and returns the time-aligned oversampled complex waveform, the demodulated bit stream, and the results of offset and drift measurements. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI assumes that the sample rate of the **input complex** waveform is exactly **samples per symbol** × the symbol rate. If this relationship does not apply to your application, use the <u>MT</u> <u>Resample (Complex Cluster)</u> VI to resample the waveform to the desired sample rate.



Note Matched filtering and/or waveform realignment performed during symbol timing recovery may lead to the apparent loss of bits. Refer to <u>Filter Delay</u> for more information about this effect. You can use the <u>MT Detect QAM</u> VI if your application requires only the demodulated bit stream output and not the **recovered complex waveform** or **measurements**.

Details



- **input complex waveform** specifies the QAM-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies time interval between data points in the acquired **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- QAM system parameters specifies parameter values defining the QAM system. Wire the QAM system parameters cluster returned

by the QAM (M) or QAM (Map) instance of the <u>MT Generate</u> <u>System Parameters</u> VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^N, where N is the number of bits per symbol. The length of the vector for the symbol(s) farthest from the origin must be 1.
- **matched filter coefficients** specifies an ordered array containing the desired matched filter coefficients. Wire the **matched filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **matched samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **QAM system parameters** cluster passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
 - **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L + 4(13 + K)]/K, where *L* is the matched filter length in taps and *K* is the number of samples per symbol. For typical values of *L* = 25 (4 samples per symbol, filter length of 6 symbols) and *K* = 4, this value equals 23.25 symbols. Therefore when **reset?** is set to TRUE, a total of 1,024 QAM symbols must be passed to the demodulator to obtain at least 1,000 symbols at the output.

This formula does not account for truncation caused by any specified **synchronization parameters**.

- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the QAM synchronization parameters cluster returned by the QAM bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.
 - Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a 90° carrier phase ambiguity.
 - expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync sequence** specifies the mapped symbol pattern used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty (default), the signal is still demodulated, but there is a phase ambiguity in the recovered symbols.
 - **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located.
 - sync indent specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the output bit stream consists of 10 data symbols, followed by the sync sequence, followed by the remaining data symbols.
- **reset?** specifies whether the VI continues demodulating using the previous iteration states. **reset?** must be TRUE whenever you want

to restart the demodulator. The VI resets on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE if the **input complex waveform** is contiguous with the **input complex waveform** from the previous iteration of this VI. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **recovered complex waveform** returns the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. The frequency offset and phase offset corrections are scalar values applied to the entire block.
 - **10** returns the time of first value in **Y** array.
 - **DEED** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **[18]** output bit stream returns the demodulated information bit stream.

- **Note** For QAM systems with more than 1 bit per symbol, such as 4-QAM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- measurements returns the results of measurements performed by the QAM demodulator.
 - **frequency offset** returns the measured carrier frequency offset, in hertz. The measured frequency offset is removed from the **recovered complex waveform**.
 - **frequency drift** returns the measured carrier frequency drift, in hertz. The measured frequency drift is not removed from the **recovered complex waveform**.
 - phase offset returns the measured phase offset, in degrees. The measured phase offset is removed from the recovered complex waveform.
- sync found index returns the symbol index within the input complex waveform where the peak correlation to the sync sequence was found. If no sync sequence is specified in the synchronization parameters cluster, this parameter returns the offset from the start of the input complex waveform to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click

the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

Bit Recovery

Use the VIs on the **Bit Recovery** palette to implement several types of digital demodulation. The VIs on this palette return only the demodulated bit stream.

Palette Object	Description
<u>MT</u> <u>Detect</u> <u>ASK</u>	Demodulates an ASK-modulated complex baseband waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
MT Detect PAM	Demodulates a pulse-amplitude-modulated (PAM) complex baseband waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
<u>MT</u> <u>Detect</u> QAM	Demodulates a QAM-modulated complex baseband waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
<u>MT</u> <u>Detect</u> <u>FSK</u>	Demodulates an FSK-modulated complex baseband waveform and returns the demodulated information bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
<u>MT</u> <u>Detect</u> <u>MSK</u>	Demodulates an MSK-modulated complex baseband waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
<u>MT</u> <u>Detect</u> <u>PSK</u>	Demodulates a PSK-modulated complex baseband waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.
MT Detect CPM	Demodulates a CPM- <u>modulated</u> complex <u>baseband</u> waveform and returns the demodulated information bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.

MT Detect ASK VI

Demodulates an ASK-modulated complex baseband waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI returns only the demodulated bit stream. Use the <u>MT Demodulate ASK</u> VI to acquire the time-aligned demodulated waveform, the demodulated bit stream, and the results of offset and drift measurements.

Note This VI assumes that the sample rate of the input complex waveform is exactly samples per symbol × the symbol rate. If this relationship does not apply to your application, use the MT Resample (Complex Cluster) VI to resample the waveform to the desired sample rate.

Details



- **input complex waveform** specifies the ASK-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the acquired **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.

ASK system parameters specifies parameter values defining the ASK system. Wire the ASK system parameters cluster from the MT Generate System Parameters VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of

samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

- **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of ASK levels in the array is 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin is 1.
- **matched filter coefficients** specifies an ordered array containing the desired matched filter coefficients. Wire the **matched filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **matched samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **ASK system parameters** cluster that is passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.

synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the **ASK synchronization parameters** cluster returned by the ASK bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.

Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a 180° carrier phase ambiguity.

EXAMPLE 2 expected sync location specifies the expected location of

the first symbol of the **sync sequence**. This value is an index to the **input complex waveform**. A value of -1 searches the entire **input complex waveform** and ignores the **sync location uncertainty** parameter.

- **sync sequence** specifies the mapped symbol pattern used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty (default), the signal is still demodulated, but there is a phase ambiguity in the recovered symbols.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located.
- sync indent specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the output bit stream consists of 10 data symbols, followed by the sync sequence, followed by the remaining data symbols.
- **TED** reset? specifies whether the VI continues demodulating using the previous iteration states. reset? must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when reset? is set to TRUE. Set reset? to FALSE if the input complex waveform is contiguous with the input complex waveform from the previous iteration of this VI. The default is TRUE.
- **flush buffers?** forces out samples from the **input complex waveform** that are delayed due to the FIR filters used in the demodulation algorithm. Set this parameter to TRUE during singleshot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the demodulated information bit stream.

- **Note** For ASK systems with more than 1 bit per symbol, such as 4-ASK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- **Sync found index** returns the symbol index within the **input complex waveform** where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

TF

status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Detect CPM VI

Demodulates a CPM-<u>modulated</u> complex <u>baseband</u> waveform and returns the demodulated information bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI returns only the demodulated bit stream. Use the <u>MT Demodulate CPM</u> VI to acquire the time-aligned demodulated waveform, the demodulated bit stream, and measurement results obtained during demodulation.

Note This VI assumes that the sample rate of the input complex waveform is exactly samples per symbol × the symbol rate. If this relationship does not apply to your application, use the MT Resample (Complex Cluster) VI to resample the waveform to the desired sample rate.

Details



- **input complex waveform** specifies the CPM-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.

CPM system parameters specifies parameter values defining the CPM system. Wire the CPM system parameters cluster from the CPM (M) or CPM (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of

samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

symbol map specifies the number of samples per symbol in the modulated output complex waveform. It should be an even number.

matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the CPM system parameters parameter in this VI.

- Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to prime the filter, hence eliminating the transient.
- **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L/2 + P/2 + 4(13 + K)]/K, where *L* is the matched filter length in taps, *P* is the pulse-shaping filter length in taps, and *K* is the number of samples per symbol. For typical values of *L* = 57, *P* = 25, and *K* = 4, this value equals 27.25 symbols. Therefore when **reset?** is set to TRUE, a total of 1,028 CPM symbols must be sent to the demodulator to obtain at least 1,000 symbols at the output. This formula does not account for truncation due to any specified **synchronization parameters.**

synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over

which to search for the sequence. Wire the **CPM synchronization parameters** cluster returned by the CPM bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.

- Note If the CPM synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a carrier phase ambiguity.
- expected sync location specifies the expected location of the first symbol of the synchronization sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
- **Sync sequence** specifies the mapped symbol pattern. Although the data type is complex, only the real portion is used. For CPM, the real portion of the mapped symbols is the frequency deviation of the symbol value, and the imaginary portion is 0. To prevent false synchronization, configure this pattern so that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty, the signal is still demodulated.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **sync indent** specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.

reset? (T) specifies an ordered array containing the desired pulseshaping coefficients. This parameter is used internally to reproduce the ideal waveform for making measurements. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter</u> <u>Coefficients</u> VI to this parameter. When generating the filter coefficients, ensure that the value of the **pulse shaping samples per symbol** parameter of the MT Generate Filter Coefficients VI is equal to the value of the **samples per symbol** element of the **CPM system parameters** cluster.

- **flush buffers?** forces out samples from the **input complex waveform** that are delayed due to the FIR filters used in the demodulation algorithm. Set this parameter to TRUE during singleshot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the demodulated information bit stream.

Note For CPM systems with more than 1 bit per symbol, such as 4-CPM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...

- **Sync found index** returns the symbol index within the **input complex waveform** where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

NI recommends using some form of pulse shaping on continuous-phase CPM-modulated signals to optimize demodulator performance. Demodulator performance under these conditions can also be improved by increasing the number of samples per symbol, but you can achieve better performance by using some form of pulse shaping.



Note This VI assumes that the input data sample rate is an integer multiple of the symbol rate. The integer multiple must match the value of the **samples per symbol** element of the **CPM system parameters**.

MT Detect FSK VI

<u>Demodulates</u> an <u>FSK</u>-modulated complex <u>baseband</u> waveform and returns the demodulated information bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI returns only the demodulated bit stream. Use the <u>MT Demodulate FSK</u> VI to acquire the time-aligned demodulated waveform, the demodulated bit stream, and measurement results obtained during demodulation.

Note This VI assumes that the sample rate of the input complex waveform is exactly samples per symbol × the symbol rate. If this relationship does not apply to your application, use the MT Resample (Complex Cluster) VI to resample the waveform to the desired sample rate.

Details



input complex waveform specifies the FSK-modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.

FSK system parameters specifies parameter values defining the FSK system. Wire the **FSK system parameters** cluster returned by the FSK (M) or FSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- Note If the symbol phase continuity element of the FSK system parameters cluster is set to discontinuous, no pulse-shaping filter can be applied, and the matched filter coefficients parameter is ignored. Refer to the FSK (M) or FSK (Map) instance of the MT Generate System Parameters VI for more information about these parameters.
- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.
- **symbol map** specifies an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of FSK levels in the array must be 2^N, where *N* is the number of bits per symbol.
- **symbol phase continuity** specifies whether the <u>phase</u> transitions between symbols are continuous.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, discontinuous phase FSK (DPFSK).

matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the FSK system parameters cluster passed to this VI.



Note When **reset?** is set to TRUE, there is a transient response of half the filter length at the start of the

demodulated signal, and the returned data is shortened by approximately half the filter length. When **reset?** is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.

 \mathbf{Q}

- **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L/2 + P/2 + 4(13 + K)]/K, where *L* is the matched filter length in taps, *P* is the pulse-shaping filter length in taps, and *K* is the number of samples per symbol. For typical values of L = 57, P = 25, and K = 4, this value equals 27.25 symbols. Therefore, when **reset?** is set to TRUE, a total of 1,028 FSK symbols must be specified to the demodulator to obtain at least 1,000 symbols at the output. This formula does not account for truncation caused by any specified **synchronization parameters**.
- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for the sequence. Wire the FSK synchronization parameters cluster returned by the FSK bit array or number array instances of the MT Generate Synchronization Parameters VI to this cluster.
 - Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a carrier phase ambiguity.
 - expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
 - **sync sequence** specifies the mapped symbol pattern.

Although the data type is complex, only the real portion is used. For FSK, the real portion represents the frequency deviation of the symbol used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. For FSK, the real portion of the mapped symbols is the <u>frequency deviation</u> of the symbol value, and the imaginary portion is 0. If this parameter is left empty, the signal is still demodulated.

- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.
- reset? specifies an ordered array containing the desired pulseshaping coefficients. This parameter is used internally to reproduce the ideal waveform for making measurements. Wire the pulse shaping filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the pulse shaping samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the FSK system parameters cluster passed to this VI.
- **IIII flush buffers?** forces out samples from the **input complex waveform** that are delayed due to the FIR filters used in the demodulation algorithm. Set this parameter to TRUE during singleshot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the demodulated information bit stream.

- **Note** For FSK systems with more than 1 bit per symbol, such as 4-FSK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- sync found index returns the symbol index within the input complex waveform where the peak correlation to the sync sequence was found. If no sync sequence is specified in the synchronization parameters cluster, this parameter returns the offset from the start of the input complex waveform to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select
Explain Error or **Explain Warning** from the shortcut menu for more information about the error.

- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Tip NI recommends using some form of pulse shaping on continuous-phase FSK-modulated signals to optimize demodulator performance. Demodulator performance under these conditions also can be improved by increasing the number of samples per symbol, but a better improvement is realized by using some form of pulse shaping.

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Detect MSK VI

Demodulates an MSK-modulated complex baseband waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI returns only the demodulated bit stream. Use the <u>MT Demodulate MSK</u> VI to acquire the time-aligned oversampled complex waveform, the ideal oversampled waveform, the demodulated bit stream, and the results of offset and drift measurements.

Note This VI assumes that the sample rate of the input complex waveform is exactly samples per symbol × the symbol rate. If this relationship does not apply to your application, use the MT Resample (Complex Cluster) VI to resample the waveform to the desired sample rate.

Details



- **input complex waveform** specifies the MSK-modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.

MSK system parameters specifies parameter values defining the MSK system. Wire the **MSK system parameters** cluster returned by the MSK instance of the <u>MT Generate System Parameters</u> VI to this cluster. Do not alter the values.

differential encoding enable specifies the status of the differential encoding.

disable (0)	Does not allow differential encoding for input symbols.
enable (1)	Allows differential encoding for input symbols. This is the default value.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - $\overline{\mathbb{N}}$

P

Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the MSK system parameters cluster passed to this VI.

Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.

Tip When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L/2 + P/2 + 4(13 + K)]/K, where *L* is the matched filter length in taps, *PP* is the pulse-shaping filter length in taps, and *K* is the number of samples per symbol. For typical values of *L* = 11, *P* = 25, and *K* = 4, this value equals 21.5 symbols. Therefore when **reset?** is set to TRUE, a total of 1,022 MSK symbols must be specified to the demodulator to obtain at least 1,000 symbols at the

output. This formula does not account for truncation due to any specified **synchronization parameters**.

- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the **MSK synchronization parameters** cluster returned by the MSK bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.
 - Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a carrier phase ambiguity.
 - expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync sequence** specifies the mapped symbol pattern used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. For MSK, the real portion of the mapped symbols is the <u>frequency deviation</u> of the symbol value, and the imaginary portion is 0. If this parameter is left empty, the signal is still demodulated.
 - **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located.
 - **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.

reset? specifies whether the VI continues demodulating using the

previous iteration states. **reset?** must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE if the **input complex waveform** is contiguous with the **input complex waveform** from the previous iteration of this VI. The default is TRUE.

flush buffers? forces out samples from the **input complex waveform** that are delayed due to the FIR filters used in the demodulation algorithm. Set this parameter to TRUE during singleshot operations and during the last iteration of continuous operations. The default is FALSE.

Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the demodulated information bit stream.

- **Sync found index** returns the symbol index within the **input complex waveform** where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Detect PAM VI

<u>Demodulates</u> a pulse-amplitude-<u>modulated</u> (PAM) complex <u>baseband</u> waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI returns only the demodulated bit stream. Use the <u>MT Demodulate PAM</u> VI to acquire the time-aligned oversampled complex waveform, the demodulated bit stream, and the results of offset and drift measurements.

Note This VI assumes that the sample rate of the input complex waveform is exactly samples per symbol × the symbol rate. If this relationship does not apply to your application, use the MT Resample (Complex Cluster) VI to resample the waveform to the desired sample rate.

Details



input complex waveform specifies the PAM-modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies time interval between data points in the acquired **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.

PAM system parameters specifies parameter values defining the PAM system. Wire the PAM system parameters cluster returned by the PAM (M) or PAM (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of

samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

- **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of PAM levels in the array is 2^{*N*}, where *N* is the number of bits per symbol. The vector length for the symbol(s) farthest from the origin is 1.
- matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the PAM system parameters cluster that is passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.

synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the **PSK synchronization parameters** cluster returned by the PSK bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.



Note If the **synchronization parameters** cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a 180° carrier phase ambiguity.

expected sync location specifies the expected location of the first symbol of the **sync sequence**. This value is an index

to the **input complex waveform**. A value of -1 searches the entire **input complex waveform** and ignores the **sync location uncertainty** parameter.

- **sync sequence** specifies the mapped symbol pattern used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty (default), the signal is still demodulated, but there is a phase ambiguity in the recovered symbols.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.
- **TED** reset? specifies whether the VI continues demodulating using the previous iteration states. reset? must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when reset? is set to TRUE. Set reset? to FALSE if the input complex waveform is contiguous with the input complex waveform from the previous iteration of this VI. The default is TRUE.

flush buffers? forces out samples from the **input complex waveform** that are delayed due to the FIR filters used in the demodulation algorithm. Set this parameter to TRUE during singleshot operations and during the last iteration of continuous operations. The default is FALSE.

Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the demodulated information bit stream.

- **Note** For PAM systems with more than 1 bit per symbol, such as 4-PAM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- **Sync found index** returns the symbol index within the **input complex waveform** where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Detect PSK VI

<u>Demodulates</u> a <u>PSK-modulated</u> complex <u>baseband</u> waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI returns only the demodulated bit stream. Use the <u>MT Demodulate PSK</u> VI to acquire the time-aligned demodulated waveform, the demodulated bit stream, and measurement results obtained during demodulation.

Note This VI assumes that the sample rate of the input complex waveform is exactly samples per symbol × the symbol rate. If this relationship does not apply to your application, use the MT Resample (Complex Cluster) VI to resample the waveform to the desired sample rate.

Details



input complex waveform specifies the PSK-modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies time interval between data points in the acquired **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.

PSK system parameters specifies parameter values defining the PSK system. Wire the PSK system parameters cluster of the PSK (M) or PSK (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of

samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

symbol map specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.

differential PSK specifies the status of the differential PSK.

disable (0) Does not allow bit sequence encoding.

enable (1) Allows bit sequence encoding.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the
	default value.

shifted Rotates the constellation by π/M each symbol.

offset Sets the modulation type to offset quadrature phaseshift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from the ideal symbol timing for I. **offset** is currently only supported for M = 4.

matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the PSK system parameters cluster passed to this VI.

- Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L + 4(13 + K)]/K, where *L* is the matched filter length in taps and *K* is the number of samples per symbol. For typical values of *L* = 25 (4 samples per symbol, filter length of 6 symbols) and *K* = 4, this value equals 23.25 symbols. Therefore when **reset?** is set to TRUE, a total of 1,024 PSK symbols must be input to the demodulator to obtain at least 1,000 symbols at the output. This formula does not account for truncation due to any specified **synchronization parameters**.

synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the **PSK synchronization parameters** cluster returned by the PSK bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.

- Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a 180° carrier phase ambiguity.
- expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
- **sync sequence** specifies the mapped symbol pattern used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data

stream. If this parameter is left empty (default), the signal is still demodulated, but there is a phase ambiguity in the recovered symbols.

- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located.
- sync indent specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the output bit stream consists of 10 data symbols, followed by the sync sequence, followed by the remaining data symbols.
- **TEP** reset? specifies whether the VI continues demodulating using the previous iteration states. reset? must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when reset? is set to TRUE. Set reset? to FALSE if the input complex waveform is contiguous with the input complex waveform from the previous iteration of this VI. The default is TRUE.
- **flush buffers?** forces out samples from the **input complex waveform** that are delayed due to the FIR filters used in the demodulation algorithm. Set this parameter to TRUE during singleshot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the demodulated information bit stream.

- **Note** For PSK systems with more than 1 bit per symbol, such as 4-PSK, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...
- sync found index returns the symbol index within the input complex waveform where the peak correlation to the sync sequence was found. If no sync sequence is specified in the synchronization parameters cluster, this parameter returns the offset from the start of the input complex waveform to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Note Offset PSK mode is only supported for 4-PSK (OQPSK), and setting **psk type** to **offset** requires some form of band-limiting filtration. Raised cosine and root-raised cosine are the only valid filter types for OQPSK demodulation. Demodulator locking performance degrades with alpha values <0.3 in the raised cosine or root-raised cosine filter design. Set the **PSK type** parameter to offset in the PSK (M) or PSK (Map) instance of <u>MT Generate</u> <u>System Parameters</u> to access this mode.

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

MT Detect QAM VI

<u>Demodulates</u> a <u>QAM-modulated</u> complex <u>baseband</u> waveform and returns the demodulated bit stream. This VI attempts to remove carrier and phase offset by locking to the carrier signal.



Note This VI returns only the demodulated bit stream. Use the <u>MT Demodulate QAM</u> VI to acquire the time-aligned demodulated waveform, the demodulated bit stream, and the results of offset and drift measurements.

Note This VI assumes that the sample rate of the input complex waveform is exactly samples per symbol × the symbol rate. If this relationship does not apply to your application, use the MT Resample (Complex Cluster) VI to resample the waveform to the desired sample rate.

Details



input complex waveform specifies the QAM-modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies time interval between data points in the acquired **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.

QAM system parameters specifies parameter values defining the QAM system. Wire the QAM system parameters cluster returned by the QAM (M) or QAM (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.
samples per symbol specifies an even, positive number of

samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbol(s) farthest from the origin must be 1.
- matched filter coefficients specifies an ordered array containing the desired matched filter coefficients. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. When generating the filter coefficients, ensure that the value of the matched samples per symbol parameter of the MT Generate Filter Coefficients VI is equal to the value of the samples per symbol element of the QAM system parameters cluster passed to this VI.
 - Note When reset? is set to TRUE, there is a transient response of half the filter length at the start of the demodulated signal, and the returned data is shortened by approximately half the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
 - **Tip** When **reset?** is set to TRUE, the number of trailing symbols that are carried over to the next iteration during demodulation is upper bounded by [L + 4(13 + K)]/K, where *L* is the matched filter length in taps and *K* is the number of samples per symbol. For typical values of L = 25 (4 samples per symbol, filter length of 6 symbols) and K = 4, this value equals 23.25 symbols. Therefore when **reset?** is set to TRUE, a total of 1,024 QAM symbols must be passed to the demodulator to obtain at least 1,000 symbols at the output. This formula does not account for truncation caused by any specified **synchronization parameters**.

synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for this sequence. Wire the **QAM synchronization parameters** cluster returned by the QAM bit array or number array instances of the <u>MT Generate Synchronization Parameters</u> VI to this cluster.

- Note If the synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a 90° carrier phase ambiguity.
- expected sync location specifies the expected location of the first symbol of the sync sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
- **sync sequence** specifies the mapped symbol pattern used to synchronize the bit stream. To prevent false synchronization, select this pattern such that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty (default), the signal is still demodulated, but there is a phase ambiguity in the recovered symbols.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.

reset? specifies whether the VI continues demodulating using the previous iteration states. **reset?** must be TRUE whenever you want to restart the demodulator. The VI resets on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE if the **input complex waveform** is contiguous with the **input complex waveform** from

the previous iteration of this VI. The default is TRUE.

- **flush buffers?** forces out samples from the **input complex waveform** that are delayed due to the FIR filters used in the demodulation algorithm. Set this parameter to TRUE during singleshot operations and during the last iteration of continuous operations. The default is FALSE.
 - Note Setting flush buffers? to TRUE destroys the internal states of the algorithms such that you cannot perform continuous processing on the signal during subsequent iterations. If flush buffers? is set to TRUE, you must set reset? to TRUE on the subsequent iteration.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns the demodulated information bit stream.

Note For QAM systems with more than 1 bit per symbol, such as 4-QAM, the symbols are converted to bits in least significant bit (LSB) first order. For example, if the detected symbols are 2,1,... the generated bits are 0,1,1,0...

- **Sync found index** returns the symbol index within the **input complex waveform** where the peak correlation to the **sync sequence** was found. If no **sync sequence** is specified in the **synchronization parameters** cluster, this parameter returns the offset from the start of the **input complex waveform** to the first complete symbol.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Successful Locking

Successful locking depends on many factors, including signal quality, modulation type, filtering parameters, and acquisition size. Locking also requires a fairly uniform distribution of symbols in the signal. The demodulator lock rate increases (and failures decrease) as the number of symbols demodulated increases. In general, you can expect to achieve a better than 95% lock when demodulating $10 \times M$ number of symbols, where *M* is 2^{bits per symbol}.

Equalization

Owning Palette: Digital

Use the VIs on the **Equalization** palette to implement several types of digital equalization.

Palette Object	Description
<u>MT ASK</u> Feedforward Equalizer	Applies adaptive feedforward software equalization, using the least-mean-squared (LMS) algorithm, to the demodulated input complex waveform . Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial conditions for the equalizer coefficients must be specified. Use the Specify Length instance when only the length of the feedforward equalizer (in symbols) must be specified.
MT_PAM Feedforward Equalizer	Applies adaptive feedforward software equalization using the least-mean-squared (LMS) algorithm to the demodulated input complex waveform . Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial conditions for the equalizer coefficients must be specified. Use the Specify Length instance when only the length of the feedforward equalizer (in symbols) must be specified.
<u>MT QAM</u> <u>Feedforward</u> <u>Equalizer</u>	Applies adaptive feedforward software equalization using the least-mean-squared (LMS) algorithm to the demodulated input complex waveform . Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial conditions for the equalizer coefficients must be specified. Use the Specify Length polymorphic instance when only the length of the feedforward equalizer (in symbols) must be specified.
<u>MT_PSK</u> <u>Feedforward</u> Equalizer	Applies adaptive feedforward software equalization using the least-mean-squared (LMS) algorithm to the demodulated input complex waveform . Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial

conditions for the equalizer coefficients must be specified.
Use the Specify Length polymorphic instance when only
the length of the feedforward equalizer (in symbols) must
be specified.

MT ASK Feedforward Equalizer VI

Applies adaptive feedforward software equalization, using the leastmean-squared (LMS) algorithm, to the demodulated **input complex waveform**. Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial conditions for the equalizer coefficients must be specified. Use the Specify Length instance when only the length of the feedforward equalizer (in symbols) must be specified.

-

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT ASK Feedforward Equalizer (LMS) (Specify coefficients)

This polymorphic instance allows you to specify equalizer coefficients or their initial conditions.



Note This VI assumes that the input sample rate is an even integer multiple of the symbol rate. The even integer multiple must match the **samples per symbol** element of the **ASK system parameters** cluster.



input complex waveform specifies the modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.

ASK system parameters specifies parameter values defining the ASK system. Wire the ASK system parameters cluster from the ASK (M) or ASK (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of ASK levels in the array is 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin is 1.

equalizer coefficients in specifies the filter coefficients for initializing the feedforward equalizer. The time span of this array (in symbols) is given by the following equation:

size of the equalizer coefficients in array / taps per symbol

For best performance and minimization of intersymbol interference, ensure that the time span (in symbols) of the equalizer coefficients filter is greater than the time span (in symbols) of the channel filter.

🕅 Note

- The equalizer coefficients in array must contain an odd number of elements.
- The values specified here are utilized by the equalizer only when **reset?** is set to TRUE or when the first instance of this VI is called. Otherwise, the VI maintains an internal buffer of the equalizer coefficients resulting from completion of the previous call.
- **training bits** specifies the binary-valued training bits that train the feedforward equalizer during the training phase when the **reset?** parameter is set to TRUE. This parameter is ignored when **reset?** is set to FALSE.

🕅 Note

- Ensure that the duration of training is long enough to allow the feedforward equalizer to converge before switching to a decision-directed adaptation.
- When working with *M*-ary modulation schemes {*M*>2, or **bits per symbol**>1} in which symbols represent groups of bits, the specified **training bits** must start on a symbol boundary.

feedforward equalizer (LMS) parameters specifies the feedforward equalization parameters. Use this cluster to define the equalizer tap spacing (symbol spaced/fractionally spaced) and convergence rate during training and steady-state.

taps per symbol specifies the tap-spacing of the **equalizer coefficients** array, relative to one symbol duration. For errorfree operation, set this parameter value to be a factor of the **samples per symbol** parameter passed from the <u>MT</u> <u>Generate System Parameters</u> VI. The default is 1.

 $\overline{\mathbb{N}}$

Note The Modulation Toolkit uses two types of equalizers that are distinguished by their tap spacing: symbol-spaced or fractionally-spaced. A *symbol-spaced* (*T-spaced*) equalizer has

taps per symbol = 1 (unity),

which means that every tap of the equalization filter spans the time duration of one symbol.

A *fractionally*-spaced (*T/N-spaced*) equalizer has

taps per symbol > 1 (unity).

For example, a *T*/2-spaced equalizer has *taps per symbol* = 2, meaning that every two taps of the equalization filter span the time duration of one symbol.

delta (training) specifies the convergence parameter for feedforward equalization during the training phase. This value adapts the feedforward equalizer coefficients for the duration of the training period when **reset?** is set to TRUE. The default is 0.01.

delta (steady state) specifies the steady-state convergence parameter for adapting the equalizer coefficients during the decision-directed phase. Use this value for adapting the feedforward equalizer coefficients when **reset?** is set to FALSE. The default is 0.01.

- **reset?** specifies whether to continue feedforward equalization using the previous iteration states. The VI always resets on first call. Set **reset?** to TRUE to reset the internal state, restart, and train the equalizer. When **reset?** is set to FALSE, the VI continues equalization from the previous iteration. This VI assumes that the **input complex waveform** is phase-continuous with the previous iteration when **reset?** is FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Output complex waveform** returns the equalized oversampled waveform generated from the feedforward equalizer as a result of adaptation. This waveform consists of the oversampled data that are compensated for the channel impulse response with zero intersymbol interference. Wire this parameter to corresponding instance of the MT Measure Quadrature Impairments VI to make modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM).
 - **IDENTIFY and SET UP: IDENTIFY and SET UP:**
 - **DEL** dt returns the time interval between data points in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **error** returns the instantaneous per symbol error in the output of the adaptive feedforward equalizer filter.
- equalizer coefficients out returns the adaptive feedforward equalizer filter coefficients when equalization is complete for the

current iteration.

- **(iii)** output bit stream returns the recovered bit stream generated from the adaptive feedforward equalizer. The recovered bit stream is phase-aligned with the output complex waveform at the equalizer output. Wire this parameter to the MT Measure Quadrature Impairments VI for performing modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM), or to the MT Calculate BER after Trigger VI for making BER measurements.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT ASK Feedforward Equalizer (LMS) (Specify length)

This polymorphic instance allows you to specify the length of the feedforward equalizer in symbols.



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- ASK system parameters specifies parameter values defining the ASK system. Wire the ASK system parameters cluster from the ASK (M) or ASK (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.
 - samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of ASK levels in the array is 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin is 1.
- equalizer length specifies the duration of the feedforward equalizer filter in symbols.
 - Note The actual length of the feedforward equalizer is given by the following equation:
N = equalizer length (symbols) × samples per symbol + 1

When **reset?** is set to TRUE, use this parameter to create the *N*-tap feedforward equalizer with default initial conditions. The default feedforward equalizer coefficients are thereby specified by the Dirac delta impulse response:

$$c(n) = \delta\left(n - \frac{N}{2}\right), n = 0 \dots N - 1$$

when reset? is set to TRUE.

training bits specifies the binary-valued training bits that train the feedforward equalizer during the training phase when the **reset?** parameter is set to TRUE. This parameter is ignored when **reset?** is set to FALSE.



- Ensure that the duration of training is long enough to allow the feedforward equalizer to converge before switching to a decision-directed adaptation.
- When working with *M*-ary modulation schemes {*M*>2, or **bits per symbol**>1} in which symbols represent groups of bits, the specified **training bits** must start on a symbol boundary.

feedforward equalizer (LMS) parameters specifies the feedforward equalization parameters. Use this cluster to define the equalizer tap spacing (symbol spaced/fractionally spaced) and convergence rate during training and steady-state.

taps per symbol specifies the tap-spacing of the equalizer coefficients array, relative to one symbol duration. For errorfree operation, set this parameter value to be a factor of the samples per symbol parameter passed from the MT Generate System Parameters VI. The default is 1.



Note The Modulation Toolkit uses two types of equalizers that are distinguished by their tap spacing: symbol-spaced or fractionally-spaced. A *symbol-spaced* (*T-spaced*) equalizer has

taps per symbol = 1 (unity),

which means that every tap of the equalization filter spans the time duration of one symbol.

A *fractionally*-spaced (*T/N-spaced*) equalizer has *taps per symbol* > 1 (unity).

For example, a *T*/2-spaced equalizer has *taps per symbol* = 2, meaning that every two taps of the equalization filter span the time duration of one symbol.

- **delta (training)** specifies the convergence parameter for feedforward equalization during the training phase. This value adapts the feedforward equalizer coefficients for the duration of the training period when **reset?** is set to TRUE. The default is 0.01.
- **delta (steady state)** specifies the steady-state convergence parameter for adapting the equalizer coefficients during the decision-directed phase. Use this value for adapting the feedforward equalizer coefficients when **reset?** is set to FALSE. The default is 0.01.
- **TEP** reset? specifies whether to continue feedforward equalization using the previous iteration states. The VI always resets on first call. Set reset? to TRUE to reset the internal state, restart, and train the equalizer. When reset? is set to FALSE, the VI continues equalization from the previous iteration. This VI assumes that the input complex waveform is phase-continuous with the previous iteration when reset? is FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

output complex waveform returns the equalized oversampled waveform generated from the feedforward equalizer as a result of adaptation. This waveform consists of the oversampled data that are compensated for the channel impulse response with zero intersymbol interference. Wire this parameter to corresponding instance of the <u>MT Measure Quadrature Impairments</u> VI to make modulation measurements such as <u>modulation error ratio (MER)</u> and <u>error vector magnitude (EVM)</u>.

- **ID** returns the trigger (start) time of the **Y** array.
- **besilt of the time interval between data points in the Y array.**
- Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **error** returns the instantaneous per symbol error in the output of the adaptive feedforward equalizer filter.
- equalizer coefficients out returns the adaptive feedforward equalizer filter coefficients when equalization is complete for the current iteration.
- **(18) output bit stream** returns the recovered bit stream generated from the adaptive feedforward equalizer. The recovered bit stream is phase-aligned with the **output complex waveform** at the equalizer output. Wire this parameter to the <u>MT Measure Quadrature</u> <u>Impairments</u> VI for performing modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM), or to the <u>MT Calculate BER after Trigger</u> VI for making BER measurements.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The **input complex waveform** is assumed to have undergone matched filtering and carrier frequency offset correction during the demodulation process (the **recovered complex waveform** returned by an MT demodulation VI meets these criteria). The adaptive equalization process ensures that the convolution of the channel filter and the equalizer coefficient filter yields a delta function that removes any intersymbol interference in the equalized **output complex waveform**.

Note This VI assumes that the sample rate is an even integer multiple of the symbol rate. The even integer multiple must match the samples per symbol element of the ASK system parameters cluster.

Because the feedforward equalizer is adaptive, the adaptation is carried out initially in a training mode and is later switched to a decision-directed mode. In the training mode, a user-supplied training bit sequence trains the equalizer to adapt to the channel conditions. In the decision-directed mode, the equalizer tap adaptation is self-directed, and the current equalizer output adapts the equalizer taps for the next iteration.

Tip For best performance during software equalization, ensure that there is an initial training period for the equalization to adapt before switching to decision-directed mode. In the absence of training data, the equalization process may fail to converge, resulting in a residual error between the equalizer output and the desired output.

The duration of the training mode is set by the number of specified **training bits** when **reset?** is set to TRUE or when the first instance of this VI is called. Training mode equalization is applied to the initial *N* symbols of the **input complex waveform**, where *N* is the number of symbols that can be mapped from the input **training bits**. Decision-directed equalization is applied to the **input complex waveform** starting at symbol *N*+1.

The equalizer is continuable, meaning that the same instance of an equalizer can be called multiple times, with each instance operating on an **input complex waveform** representing a subsequent frame of the same data stream. To operate in this manner, call the equalizer in a loop

with the first iteration setting **reset?** to TRUE and subsequent iterations setting **reset?** to FALSE.

The equalizer can also be applied to a discontinuous data stream, where the same instance of the equalizer operates on **input complex waveforms** that do not maintain phase continuity from one to the next. To operate in this manner, call the same instance of the equalizer in a loop with **reset?** set to TRUE for all iterations.

The Specify Coefficients instance of this VI allows continuous equalization on discontinuous data. If you are working with discontinuous data and know that your input channel is invariant enough to potentially benefit from continuous equalization, wire **equalizer coefficients out** to the **equalizer coefficients in** parameter. To do so, call the equalizer in a loop with **reset?** set to TRUE and use a feedback node or shift register to wire the output of a previous iteration as the input for the next iteration. This technique also allows continuous training of the equalizer; if supplied with appropriate training bits, the equalizer retrains with every iteration.

The following flow chart illustrates demodulator-to-equalizer data flow.



MT PAM Feedforward Equalizer VI

Applies adaptive feedforward software equalization using the least-meansquared (LMS) algorithm to the demodulated **input complex waveform**. Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial conditions for the equalizer coefficients must be specified. Use the Specify Length instance when only the length of the feedforward equalizer (in symbols) must be specified.

▼

Use the pull-down menu to select an instance of this VI.

Select an instance

<u>Details</u>

MT PAM Feedforward Equalizer (LMS) (Specify coefficients)

This polymorphic instance allows you to specify equalizer coefficients or their initial conditions.



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- **PAM system parameters** specifies parameter values defining the PAM system. Wire the **PAM system parameters** cluster returned by either the PAM (M) or PAM (Map) instance of the <u>MT Generate</u> <u>System Parameters</u> VI to this cluster.
 - samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of states in the array must be 2^N, where N is the number of bits per symbol.
- equalizer coefficients in specifies the filter coefficients for initializing the feedforward equalizer. The time span of this array (in symbols) is given by the following equation:

size of the equalizer coefficients in array / taps per symbol

For best performance and minimization of intersymbol interference,

ensure that the time span (in symbols) of the equalizer coefficients filter is greater than the time span (in symbols) of the channel filter.

- 🔊 Note
- The **equalizer coefficients in** array must contain an odd number of elements.
- The values specified here are utilized by the equalizer only when **reset?** is set to TRUE or when the first instance of this VI is called. Otherwise, the VI maintains an internal buffer of the equalizer coefficients resulting from completion of the previous call.

training bits specifies the binary-valued training bits that train the feedforward equalizer during the training phase when the **reset?** parameter is set to TRUE. This parameter is ignored when **reset?** is set to FALSE.



- Ensure that the duration of training is long enough to allow the feedforward equalizer to converge before switching to a decision-directed adaptation.
- When working with *M*-ary modulation schemes {*M*>2, or **bits per symbol**>1} in which symbols represent groups of bits, the specified **training bits** must start on a symbol boundary.
- **feedforward equalizer (LMS) parameters** specifies the feedforward equalization parameters. Use this cluster to define the equalizer tap spacing (symbol spaced/fractionally spaced) and convergence rate during training and steady-state.
 - taps per symbol specifies the tap-spacing of the equalizer coefficients array, relative to one symbol duration. For errorfree operation, set this parameter value to be a factor of the samples per symbol parameter passed from the MT Generate System Parameters VI. The default is 1.



Note The Modulation Toolkit uses two types of equalizers that are distinguished by their tap spacing: symbol-spaced or fractionally-spaced. A *symbol-*

spaced (T-spaced) equalizer has

taps per symbol = 1 (unity),

which means that every tap of the equalization filter spans the time duration of one symbol.

A *fractionally*-spaced (*T/N-spaced*) equalizer has

taps per symbol > 1 (unity).

For example, a *T*/2-spaced equalizer has *taps per symbol* = 2, meaning that every two taps of the equalization filter span the time duration of one symbol.

- delta (training) specifies the convergence parameter for feedforward equalization during the training phase. This value adapts the feedforward equalizer coefficients for the duration of the training period when **reset?** is set to TRUE. The default is 0.01.
- **delta (steady state)** specifies the steady-state convergence parameter for adapting the equalizer coefficients during the decision-directed phase. Use this value for adapting the feedforward equalizer coefficients when **reset?** is set to FALSE. The default is 0.01.
- reset? specifies whether to continue feedforward equalization using the previous iteration states. The VI always resets on first call. Set reset? to TRUE to reset the internal state, restart, and train the equalizer. When reset? is set to FALSE, the VI continues equalization from the previous iteration. This VI assumes that the input complex waveform is phase-continuous with the previous iteration when reset? is FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **WETT output complex waveform** returns the equalized oversampled waveform generated from the feedforward equalizer as a result of adaptation. This waveform consists of the oversampled data that are compensated for the channel impulse response with zero intersymbol interference. Wire this parameter to corresponding instance of the MT Measure Quadrature Impairments VI to make modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM).
 - **ID** returns the trigger (start) time of the **Y** array.
 - **DEL** dt returns the time interval between data points in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **error** returns the instantaneous per symbol error in the output of the adaptive feedforward equalizer filter.
- equalizer coefficients out returns the adaptive feedforward equalizer filter coefficients when equalization is complete for the current iteration.
- **(III) output bit stream** returns the recovered bit stream generated from the adaptive feedforward equalizer. The recovered bit stream is phase-aligned with the **output complex waveform** at the equalizer output. Wire this parameter to the <u>MT Measure Quadrature</u> <u>Impairments</u> VI for performing modulation measurements such as

modulation error ratio (MER) and error vector magnitude (EVM), or to the MT Calculate BER after Trigger VI for making BER measurements.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT PAM Feedforward Equalizer (LMS) (Specify length)

This polymorphic instance allows you to specify the length of the feedforward equalizer in symbols.



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- PAM system parameters specifies parameter values defining the PAM system. Wire the PAM system parameters cluster returned by either the PAM (M) or PAM (Map) instance of the <u>MT Generate</u> <u>System Parameters</u> VI to this cluster.
 - **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of states in the array must be 2^N, where N is the number of bits per symbol.
- **equalizer length** specifies the duration of the feedforward equalizer filter in symbols.
 - $\overline{\mathbb{N}}$
- **Note** The actual length of the feedforward equalizer is given by the following equation:

N = equalizer length (symbols) × samples per symbol + 1

When **reset?** is set to TRUE, use this parameter to create the *N*-tap feedforward equalizer with default initial conditions. The default feedforward equalizer coefficients are thereby specified by the Dirac delta impulse response:

$$c(n) = \delta\left(n - \frac{N}{2}\right), n = 0 \dots N - 1$$

when reset? is set to TRUE.

training bits specifies the binary-valued training bits that train the feedforward equalizer during the training phase when the **reset?** parameter is set to TRUE. This parameter is ignored when **reset?** is set to FALSE.



- Ensure that the duration of training is long enough to allow the feedforward equalizer to converge before switching to a decision-directed adaptation.
- When working with *M*-ary modulation schemes {*M*>2, or **bits per symbol**>1} in which symbols represent groups of bits, the specified **training bits** must start on a symbol boundary.

feedforward equalizer (LMS) parameters specifies the feedforward equalization parameters. Use this cluster to define the equalizer tap spacing (symbol spaced/fractionally spaced) and convergence rate during training and steady-state.

taps per symbol specifies the tap-spacing of the equalizer coefficients array, relative to one symbol duration. For errorfree operation, set this parameter value to be a factor of the samples per symbol parameter passed from the MT Generate System Parameters VI. The default is 1.



Note The Modulation Toolkit uses two types of equalizers that are distinguished by their tap spacing: symbol-spaced or fractionally-spaced. A *symbol-spaced* (*T-spaced*) equalizer has

taps per symbol = 1 (unity),

which means that every tap of the equalization filter spans the time duration of one symbol.

A *fractionally*-spaced (*T/N-spaced*) equalizer has *taps per symbol* > 1 (unity).

For example, a *T*/2-spaced equalizer has *taps per symbol* = 2, meaning that every two taps of the equalization filter span the time duration of one symbol.

- **delta (training)** specifies the convergence parameter for feedforward equalization during the training phase. This value adapts the feedforward equalizer coefficients for the duration of the training period when **reset?** is set to TRUE. The default is 0.01.
- **delta (steady state)** specifies the steady-state convergence parameter for adapting the equalizer coefficients during the decision-directed phase. Use this value for adapting the feedforward equalizer coefficients when **reset?** is set to FALSE. The default is 0.01.
- **TEP** reset? specifies whether to continue feedforward equalization using the previous iteration states. The VI always resets on first call. Set reset? to TRUE to reset the internal state, restart, and train the equalizer. When reset? is set to FALSE, the VI continues equalization from the previous iteration. This VI assumes that the input complex waveform is phase-continuous with the previous iteration when reset? is FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

output complex waveform returns the equalized oversampled waveform generated from the feedforward equalizer as a result of adaptation. This waveform consists of the oversampled data that are compensated for the channel impulse response with zero intersymbol interference. Wire this parameter to corresponding instance of the MT Measure Quadrature Impairments VI to make modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM).

- **ID** returns the trigger (start) time of the **Y** array.
- **DEL** dt returns the time interval between data points in the Y array.
- Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **error** returns the instantaneous per symbol error in the output of the adaptive feedforward equalizer filter.
- equalizer coefficients out returns the adaptive feedforward equalizer filter coefficients when equalization is complete for the current iteration.
- **(iii)** output bit stream returns the recovered bit stream generated from the adaptive feedforward equalizer. The recovered bit stream is phase-aligned with the output complex waveform at the equalizer output. Wire this parameter to the MT Measure Quadrature Impairments VI for performing modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM), or to the MT Calculate BER after Trigger VI for making BER measurements.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The **input complex waveform** is assumed to have undergone matched filtering and carrier frequency offset correction during the demodulation process (the **recovered complex waveform** returned by an MT demodulation VI meets these criteria). The adaptive equalization process ensures that the convolution of the channel filter and the equalizer coefficient filter yields a delta function that removes any intersymbol interference in the equalized **output complex waveform**.

Note This VI assumes that the sample rate is an even integer multiple of the symbol rate. The even integer multiple must match the samples per symbol element of the PAM system parameters cluster.

Because the feedforward equalizer is adaptive, the adaptation is carried out initially in a training mode and is later switched to a decision-directed mode. In the training mode, a user-supplied training bit sequence trains the equalizer to adapt to the channel conditions. In the decision-directed mode, the equalizer tap adaptation is self-directed, and the current equalizer output adapts the equalizer taps for the next iteration.

Tip For best performance during software equalization, ensure that there is an initial training period for the equalization to adapt before switching to decision-directed mode. In the absence of training data, the equalization process may fail to converge, resulting in a residual error between the equalizer output and the desired output.

The duration of the training mode is set by the number of specified **training bits** when **reset?** is set to TRUE, or when the first instance of this VI is called. Training mode equalization is applied to the initial *N* symbols of the **input complex waveform**, where *N* is the number of symbols that can be mapped from the input **training bits**. Decision-directed equalization is applied to the **input complex waveform** starting at symbol *N*+1.

The equalizer is continuable, meaning that the same instance of an equalizer can be called multiple times, with each instance operating on an **input complex waveform** representing a subsequent frame of the same data stream. To operate in this manner, call the equalizer in a loop

with the first iteration setting **reset?** to TRUE and subsequent iterations setting **reset?** to FALSE.

The equalizer can also be applied to a discontinuous data stream, in which the same instance of the equalizer operates on **input complex waveforms** that do not maintain phase continuity from one to the next. To operate in this manner, call the same instance of the equalizer in a loop with **reset?** set to TRUE for all iterations.

The Specify Coefficients instance of this VI allows continuous equalization on discontinuous data. If you are working with discontinuous data and know that your input channel is invariant enough to potentially benefit from continuous equalization, wire **equalizer coefficients out** to the **equalizer coefficients in** parameter. To do so, call the equalizer in a loop with **reset?** set to TRUE and use a feedback node or shift register to wire the output of a previous iteration as the input for the next iteration. This technique also allows continuous training of the equalizer; if supplied with appropriate training bits, the equalizer retrains with every iteration.

The following flow chart illustrates demodulator-to-equalizer data flow.



MT PSK Feedforward Equalizer VI

Applies adaptive feedforward software equalization using the least-meansquared (LMS) algorithm to the demodulated **input complex waveform**. Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial conditions for the equalizer coefficients must be specified. Use the Specify Length polymorphic instance when only the length of the feedforward equalizer (in symbols) must be specified.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

<u>Details</u>

MT PSK Feedforward Equalizer (LMS) (Specify coefficients)

This polymorphic instance allows you to specify equalizer coefficients or their initial conditions.



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- PSK system parameters specifies parameter values defining the PSK system. Wire the PSK system parameters cluster from the PSK (M) or PSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.
 - samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of states in the array must be 2^N, where N is the number of bits per symbol.
 - differential PSK specifies how the PSK modulation represents symbols. Differential operation is used to implement PSK formats such as differential quadrature PSK (DQPSK) and π/4-DQPSK.

disable (0) Symbols are represented as constellation points.

	This is the default value.
enable (1)	Symbols are represented as the transitions between constellation points.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase- shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from the ideal symbol timing for I. offset is currently only supported for $M = 4$.

equalizer coefficients in specifies the filter coefficients for initializing the feedforward equalizer. The time span of this array (in symbols) is given by the following equation:

size of the equalizer coefficients in array / taps per symbol

For best performance and minimization of intersymbol interference, ensure that the time span (in symbols) of the equalizer coefficients filter is greater than the time span (in symbols) of the channel filter.

🕅 Note

- The equalizer coefficients in array must contain an odd number of elements.
- The values specified here are utilized by the equalizer only when **reset?** is set to TRUE or when the first instance of this VI is called. Otherwise, the VI maintains an internal buffer of the equalizer coefficients resulting from completion of the previous call.

training bits specifies the binary-valued training bits that train the feedforward equalizer during the training phase when the **reset?**

parameter is set to TRUE. This parameter is ignored when **reset?** is set to FALSE.

- 🔊 Note
- Ensure that the duration of training is long enough to allow the feedforward equalizer to converge before switching to a decision-directed adaptation.
- When working with *M*-ary modulation schemes {*M*>2, or **bits per symbol**>1} in which symbols represent groups of bits, the specified **training bits** must start on a symbol boundary.

feedforward equalizer (LMS) parameters specifies the feedforward equalization parameters. Use this cluster to define the equalizer tap spacing (symbol spaced/fractionally spaced) and convergence rate during training and steady-state.

- taps per symbol specifies the tap-spacing of the equalizer coefficients array, relative to one symbol duration. For errorfree operation, set this parameter value to be a factor of the samples per symbol parameter passed from the <u>MT</u> <u>Generate System Parameters</u> VI. The default is 1.

Note The Modulation Toolkit uses two types of equalizers that are distinguished by their tap spacing: symbol-spaced or fractionally-spaced. A *symbol-spaced* (*T-spaced*) equalizer has

taps per symbol = 1 (unity),

which means that every tap of the equalization filter spans the time duration of one symbol.

A *fractionally*-spaced (*T/N-spaced*) equalizer has

taps per symbol > 1 (unity).

For example, a *T*/2-spaced equalizer has *taps per symbol* = 2, meaning that every two taps of the equalization filter span the time duration of one symbol.

delta (training) specifies the convergence parameter for feedforward equalization during the training phase. This value adapts the feedforward equalizer coefficients for the duration of the training period when **reset?** is set to TRUE. The default is 0.01.

- **delta (steady state)** specifies the steady-state convergence parameter for adapting the equalizer coefficients during the decision-directed phase. Use this value for adapting the feedforward equalizer coefficients when **reset?** is set to FALSE. The default is 0.01.
- **reset?** specifies whether to continue feedforward equalization using the previous iteration states. The VI always resets on first call. Set **reset?** to TRUE to reset the internal state, restart, and train the equalizer. When **reset?** is set to FALSE, the VI continues equalization from the previous iteration. This VI assumes that the **input complex waveform** is phase-continuous with the previous iteration when **reset?** is FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the equalized oversampled waveform generated from the feedforward equalizer as a result of

adaptation. This waveform consists of the oversampled data that are compensated for the channel impulse response with zero intersymbol interference. Wire this parameter to corresponding instance of the <u>MT Measure Quadrature Impairments</u> VI to make modulation measurements such as <u>modulation error ratio (MER)</u> and <u>error vector magnitude (EVM)</u>.

- **ID** returns the trigger (start) time of the **Y** array.
- **besilt of the time interval between data points in the Y array.**
- Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **error** returns the instantaneous per symbol error in the output of the adaptive feedforward equalizer filter.
- equalizer coefficients out returns the adaptive feedforward equalizer filter coefficients when equalization is complete for the current iteration.
- **(iii)** output bit stream returns the recovered bit stream generated from the adaptive feedforward equalizer. The recovered bit stream is phase-aligned with the output complex waveform at the equalizer output. Wire this parameter to the MT Measure Quadrature Impairments VI for performing modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM), or to the MT Calculate BER after Trigger VI for making BER measurements.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain**

Warning from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT PSK Feedforward Equalizer (LMS) (Specify length)

This polymorphic instance allows you to specify the length of the feedforward equalizer in symbols.



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- PSK system parameters specifies parameter values defining the PSK system. Wire the PSK system parameters cluster from the PSK (M) or PSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.
 - samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of states in the array must be 2^N, where N is the number of bits per symbol.
 - differential PSK specifies how the PSK modulation represents symbols. Differential operation is used to implement PSK formats such as differential quadrature PSK (DQPSK) and π/4-DQPSK.

disable (0) Symbols are represented as constellation points.

	This is the default value.
enable (1)	Symbols are represented as the transitions between constellation points.
	between constellation points.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase- shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from the ideal symbol timing for I. offset is currently only supported for $M = 4$.

- equalizer length specifies the duration of the feedforward equalizer filter in symbols.
 - 🔊 No

Note The actual length of the feedforward equalizer is given by the following equation:

 $N = equalizer length (symbols) \times samples per symbol + 1$

When **reset?** is set to TRUE, use this parameter to create the *N*-tap feedforward equalizer with default initial conditions. The default feedforward equalizer coefficients are thereby specified by the Dirac delta impulse response:

$$c(n) = \delta\left(n - \frac{N}{2}\right), n = 0 \dots N-1$$

when reset? is set to TRUE.

training bits specifies the binary-valued training bits that train the feedforward equalizer during the training phase when the **reset?** parameter is set to TRUE. This parameter is ignored when **reset?** is set to FALSE.



- Ensure that the duration of training is long enough to allow the feedforward equalizer to converge before switching to a decision-directed adaptation.
- When working with *M*-ary modulation schemes {*M*>2, or **bits per symbol**>1} in which symbols represent groups of bits, the specified **training bits** must start on a symbol boundary.

feedforward equalizer (LMS) parameters specifies the feedforward equalization parameters. Use this cluster to define the equalizer tap spacing (symbol spaced/fractionally spaced) and convergence rate during training and steady-state.

- taps per symbol specifies the tap-spacing of the equalizer coefficients array, relative to one symbol duration. For errorfree operation, set this parameter value to be a factor of the samples per symbol parameter passed from the MT Generate System Parameters VI. The default is 1.
 - $\overline{\mathbb{N}}$

Note The Modulation Toolkit uses two types of equalizers that are distinguished by their tap spacing: symbol-spaced or fractionally-spaced. A *symbol-spaced* (*T-spaced*) equalizer has

taps per symbol = 1 (unity),

which means that every tap of the equalization filter spans the time duration of one symbol.

A *fractionally*-spaced (*T/N-spaced*) equalizer has

taps per symbol > 1 (unity).

For example, a T/2-spaced equalizer has *taps per* symbol = 2, meaning that every two taps of the equalization filter span the time duration of one symbol.

delta (training) specifies the convergence parameter for feedforward equalization during the training phase. This value adapts the feedforward equalizer coefficients for the duration of the training period when **reset?** is set to TRUE. The default is 0.01.

DBL

delta (steady state) specifies the steady-state convergence parameter for adapting the equalizer coefficients during the decision-directed phase. Use this value for adapting the feedforward equalizer coefficients when **reset?** is set to FALSE. The default is 0.01.

- **reset?** specifies whether to continue feedforward equalization using the previous iteration states. The VI always resets on first call. Set **reset?** to TRUE to reset the internal state, restart, and train the equalizer. When **reset?** is set to FALSE, the VI continues equalization from the previous iteration. This VI assumes that the **input complex waveform** is phase-continuous with the previous iteration when **reset?** is FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the equalized oversampled waveform generated from the feedforward equalizer as a result of adaptation. This waveform consists of the oversampled data that are compensated for the channel impulse response with zero intersymbol interference. Wire this parameter to corresponding

instance of the <u>MT Measure Quadrature Impairments</u> VI to make modulation measurements such as <u>modulation error ratio (MER)</u> and <u>error vector magnitude (EVM)</u>.

- **ID** returns the trigger (start) time of the **Y** array.
- **DEED** dt returns the time interval between data points in the Y array.
- Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- **error** returns the instantaneous per symbol error in the output of the adaptive feedforward equalizer filter.
- equalizer coefficients out returns the adaptive feedforward equalizer filter coefficients when equalization is complete for the current iteration.
- **(III) output bit stream** returns the recovered bit stream generated from the adaptive feedforward equalizer. The recovered bit stream is phase-aligned with the **output complex waveform** at the equalizer output. Wire this parameter to the <u>MT Measure Quadrature</u> <u>Impairments</u> VI for performing modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM), or to the <u>MT Calculate BER after Trigger</u> VI for making BER measurements.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

abc

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The **input complex waveform** is assumed to have undergone matched filtering and carrier frequency offset correction during the demodulation process (the **recovered complex waveform** returned by an MT demodulation VI meets these criteria). The adaptive equalization process ensures that the convolution of the channel filter and the equalizer coefficient filter yields a delta function, thereby removing any intersymbol interference in the equalized output complex waveform.



Because the feedforward equalizer is adaptive, the adaptation is carried out initially in a training mode and is later switched to a decision-directed mode. In the training mode, a user-supplied training bit sequence trains the equalizer to adapt to the channel conditions. In the decision-directed mode, the equalizer tap adaptation is self-directed, where the current equalizer output adapts the equalizer taps for the next iteration.

Tip For best performance during software equalization, ensure that there is an initial training period for the equalization to adapt before switching to decision-directed mode. In the absence of training data, the equalization process may fail to converge, resulting in a residual error between the equalizer output and the desired output.

The duration of the training mode is set by the number of specified **training bits** when **reset?** is set to TRUE or when the first instance of this VI is called. Training mode equalization is applied to the initial *N* symbols of the **input complex waveform**, where *N* is the number of symbols that can be mapped from the input **training bits**. Decision-directed equalization is applied to the **input complex waveform** starting at symbol *N*+1.

The equalizer is continuable, meaning that the same instance of an equalizer can be called multiple times, with each instance operating on an **input complex waveform** representing a subsequent frame of the same data stream. To operate in this manner, you would call the

equalizer in a loop with the first iteration setting **reset?** to TRUE and subsequent iterations setting **reset?** to FALSE.

The equalizer can also be applied to a discontinuous data stream, where the same instance of the equalizer operates on **input complex waveforms** that do not maintain phase continuity from one to the next. To operate in this manner, call the same instance of the equalizer in a loop with **reset?** set to TRUE for all iterations.

The Specify Coefficients instance of this VI allows continuous equalization on discontinuous data. If you are working with discontinuous data and know that your input channel is invariant enough to potentially benefit from continuous equalization, wire **equalizer coefficients out** to the **equalizer coefficients in** parameter. To do so, call the equalizer in a loop with **reset?** set to TRUE and use a feedback node or shift register to wire the output of a previous iteration as the input for the next iteration. This technique also allows continuous training of the equalizer; if supplied with appropriate training bits, the equalizer retrains with every iteration.

The following flow chart illustrates demodulator-to-equalizer data flow.



MT QAM Feedforward Equalizer VI

Applies adaptive feedforward software equalization using the least-meansquared (LMS) algorithm to the demodulated **input complex waveform**. Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial conditions for the equalizer coefficients must be specified. Use the Specify Length polymorphic instance when only the length of the feedforward equalizer (in symbols) must be specified.

▼

Use the pull-down menu to select an instance of this VI.

Select an instance

<u>Details</u>

MT QAM Feedforward Equalizer (LMS) (Specify coefficients)

This polymorphic instance allows you to specify equalizer coefficients or their initial conditions.



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- QAM system parameters specifies parameter values defining the QAM system. Wire the QAM system parameters cluster returned by the QAM (M) or QAM (Map) instance of the <u>MT Generate</u> System Parameters VI to this cluster.
 - samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of states in the array must be 2^N, where N is the number of bits per symbol.
- equalizer coefficients in specifies the filter coefficients for initializing the feedforward equalizer. The time span of this array (in symbols) is given by the following equation:

size of the equalizer coefficients in array / taps per symbol
For best performance and minimization of intersymbol interference, ensure that the time span (in symbols) of the equalizer coefficients filter is greater than the time span (in symbols) of the channel filter.

🕅 Note

- The **equalizer coefficients in** array must contain an odd number of elements.
- The values specified here are utilized by the equalizer only when **reset?** is set to TRUE or when the first instance of this VI is called. Otherwise, the VI maintains an internal buffer of the equalizer coefficients resulting from completion of the previous call.

training bits specifies the binary-valued training bits that train the feedforward equalizer during the training phase when the **reset?** parameter is set to TRUE. This parameter is ignored when **reset?** is set to FALSE.



- Ensure that the duration of training is long enough to allow the feedforward equalizer to converge before switching to a decision-directed adaptation.
- When working with *M*-ary modulation schemes {*M*>2, or **bits per symbol**>1} in which symbols represent groups of bits, the specified **training bits** must start on a symbol boundary.

feedforward equalizer (LMS) parameters specifies the feedforward equalization parameters. Use this cluster to define the equalizer tap spacing (symbol spaced/fractionally spaced) and convergence rate during training and steady-state.

taps per symbol specifies the tap-spacing of the equalizer coefficients array, relative to one symbol duration. For errorfree operation, set this parameter value to be a factor of the samples per symbol parameter passed from the MT Generate System Parameters VI. The default is 1.



Note The Modulation Toolkit uses two types of equalizers that are distinguished by their tap spacing:

symbol-spaced or fractionally-spaced. A symbol-spaced (*T-spaced*) equalizer has

taps per symbol = 1 (unity),

which means that every tap of the equalization filter spans the time duration of one symbol.

A fractionally-spaced (T/N-spaced) equalizer has

taps per symbol > 1 (unity).

For example, a *T*/2-spaced equalizer has *taps per symbol* = 2, meaning that every two taps of the equalization filter span the time duration of one symbol.

- **delta (training)** specifies the convergence parameter for feedforward equalization during the training phase. This value adapts the feedforward equalizer coefficients for the duration of the training period when **reset?** is set to TRUE. The default is 0.01.
- **delta (steady state)** specifies the steady-state convergence parameter for adapting the equalizer coefficients during the decision-directed phase. Use this value for adapting the feedforward equalizer coefficients when **reset?** is set to FALSE. The default is 0.01.
- reset? specifies whether to continue feedforward equalization using the previous iteration states. The VI always resets on first call. Set reset? to TRUE to reset the internal state, restart, and train the equalizer. When reset? is set to FALSE, the VI continues equalization from the previous iteration. This VI assumes that the input complex waveform is phase-continuous with the previous iteration when reset? is FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the equalized oversampled waveform generated from the feedforward equalizer as a result of adaptation. This waveform consists of the oversampled data that are compensated for the channel impulse response with zero intersymbol interference. Wire this parameter to corresponding instance of the <u>MT Measure Quadrature Impairments</u> VI to make modulation measurements such as <u>modulation error ratio (MER)</u> and <u>error vector magnitude (EVM)</u>.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **DEED** dt returns the time interval between data points in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- **error** returns the instantaneous per symbol error in the output of the adaptive feedforward equalizer filter.
- equalizer coefficients out returns the adaptive feedforward equalizer filter coefficients when equalization is complete for the current iteration.
- **(III) output bit stream** returns the recovered bit stream generated from the adaptive feedforward equalizer. The recovered bit stream is phase-aligned with the **output complex waveform** at the equalizer output. Wire this parameter to the <u>MT Measure Quadrature</u>

Impairments VI for performing modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM), or to the MT Calculate BER after Trigger VI for making BER measurements.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT QAM Feedforward Equalizer (LMS) (Specify length)

This polymorphic instance allows you to specify the length of the feedforward equalizer in symbols.



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- QAM system parameters specifies parameter values defining the QAM system. Wire the QAM system parameters cluster returned by the QAM (M) or QAM (Map) instance of the <u>MT Generate</u> <u>System Parameters</u> VI to this cluster.
 - **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of states in the array must be 2^N, where N is the number of bits per symbol.
- equalizer length specifies the duration of the feedforward equalizer filter in symbols.
 - $\overline{\mathbb{N}}$
- **Note** The actual length of the feedforward equalizer is given by the following equation:

N = equalizer length (symbols) × samples per symbol + 1

When **reset?** is set to TRUE, use this parameter to create the *N*-tap feedforward equalizer with default initial conditions. The default feedforward equalizer coefficients are thereby specified by the Dirac delta impulse response:

$$c(n) = \delta\left(n - \frac{N}{2}\right), n = 0 \dots N - 1$$

when reset? is set to TRUE.

training bits specifies the binary-valued training bits that train the feedforward equalizer during the training phase when the **reset?** parameter is set to TRUE. This parameter is ignored when **reset?** is set to FALSE.



- Ensure that the duration of training is long enough to allow the feedforward equalizer to converge before switching to a decision-directed adaptation.
- When working with *M*-ary modulation schemes {*M*>2, or **bits per symbol**>1} in which symbols represent groups of bits, the specified **training bits** must start on a symbol boundary.

feedforward equalizer (LMS) parameters specifies the feedforward equalization parameters. Use this cluster to define the equalizer tap spacing (symbol spaced/fractionally spaced) and convergence rate during training and steady-state.

taps per symbol specifies the tap-spacing of the equalizer coefficients array, relative to one symbol duration. For errorfree operation, set this parameter value to be a factor of the samples per symbol parameter passed from the MT Generate System Parameters VI. The default is 1.



Note The Modulation Toolkit uses two types of equalizers that are distinguished by their tap spacing: symbol-spaced or fractionally-spaced. A *symbol-spaced* (*T-spaced*) equalizer has

taps per symbol = 1 (unity),

which means that every tap of the equalization filter spans the time duration of one symbol.

A *fractionally*-spaced (*T/N-spaced*) equalizer has *taps per symbol* > 1 (unity).

For example, a *T*/2-spaced equalizer has *taps per symbol* = 2, meaning that every two taps of the equalization filter span the time duration of one symbol.

- **delta (training)** specifies the convergence parameter for feedforward equalization during the training phase. This value adapts the feedforward equalizer coefficients for the duration of the training period when **reset?** is set to TRUE. The default is 0.01.
- **delta (steady state)** specifies the steady-state convergence parameter for adapting the equalizer coefficients during the decision-directed phase. Use this value for adapting the feedforward equalizer coefficients when **reset?** is set to FALSE. The default is 0.01.
- **TEP** reset? specifies whether to continue feedforward equalization using the previous iteration states. The VI always resets on first call. Set reset? to TRUE to reset the internal state, restart, and train the equalizer. When reset? is set to FALSE, the VI continues equalization from the previous iteration. This VI assumes that the input complex waveform is phase-continuous with the previous iteration when reset? is FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

output complex waveform returns the equalized oversampled waveform generated from the feedforward equalizer as a result of adaptation. This waveform consists of the oversampled data that are compensated for the channel impulse response with zero intersymbol interference. Wire this parameter to corresponding instance of the MT Measure Quadrature Impairments VI to make modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM).

- **ID** returns the trigger (start) time of the **Y** array.
- **DEL** dt returns the time interval between data points in the Y array.
- Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **error** returns the instantaneous per symbol error in the output of the adaptive feedforward equalizer filter.
- equalizer coefficients out returns the adaptive feedforward equalizer filter coefficients when equalization is complete for the current iteration.
- **(iii)** output bit stream returns the recovered bit stream generated from the adaptive feedforward equalizer. The recovered bit stream is phase-aligned with the output complex waveform at the equalizer output. Wire this parameter to the MT Measure Quadrature Impairments VI for performing modulation measurements such as modulation error ratio (MER) and error vector magnitude (EVM), or to the MT Calculate BER after Trigger VI for making BER measurements.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The **input complex waveform** is assumed to have undergone matched filtering and carrier frequency offset correction during the demodulation process (the **recovered complex waveform** returned by an MT demodulation VI meets these criteria). The adaptive equalization process ensures that the convolution of the channel filter and the equalizer coefficient filter yields a delta function, thereby removing any intersymbol interference in the equalized **output complex waveform**.

Note This VI assumes that the sample rate is an even integer multiple of the symbol rate. The even integer multiple must match the samples per symbol element of the QAM system parameters cluster.

Because the feedforward equalizer is adaptive, the adaptation is carried out initially in a training mode and is later switched to a decision-directed mode. In the training mode, a user-supplied training bit sequence trains the equalizer to adapt to the channel conditions. In the decision-directed mode, the equalizer tap adaptation is self-directed, where the current equalizer output adapts the equalizer taps for the next iteration.

Tip For best performance during software equalization, ensure that there is an initial training period for the equalization to adapt before switching to decision-directed mode. In the absence of training data, the equalization process may fail to converge, resulting in a residual error between the equalizer output and the desired output.

The duration of the training mode is set by the number of specified **training bits** when **reset?** is set to TRUE or when the first instance of this VI is called. Training mode equalization is applied to the initial *N* symbols of the **input complex waveform**, where *N* is the number of symbols that can be mapped from the input **training bits**. Decision-directed equalization is applied to the **input complex waveform** starting at symbol *N*+1.

The equalizer is continuable, meaning that the same instance of an equalizer can be called multiple times, with each instance operating on an **input complex waveform** representing a subsequent frame of the same data stream. To operate in this manner, call the equalizer in a loop

with the first iteration setting **reset?** to TRUE and subsequent iterations setting **reset?** to FALSE.

The equalizer can also be applied to a discontinuous data stream, where the same instance of the equalizer operates on **input complex waveforms** that do not maintain phase continuity from one to the next. To operate in this manner, call the same instance of the equalizer in a loop with **reset?** set to TRUE for all iterations.

The Specify Coefficients instance of this VI allows continuous equalization on discontinuous data. If you are working with discontinuous data and know that your input channel is invariant enough to potentially benefit from continuous equalization, wire **equalizer coefficients out** to the **equalizer coefficients in** parameter. To do so, call the equalizer in a loop with **reset?** set to TRUE and use a feedback node or shift register to wire the output of a previous iteration as the input for the next iteration. This technique also allows continuous training of the equalizer; if supplied with appropriate training bits, the equalizer retrains with every iteration.

The following flow chart illustrates demodulator-to-equalizer data flow.



Utilities

Owning Palette: <u>Digital</u>

Use the VIs on the **Utilities** palette to perform tasks such as applying filters, generating synchronization parameters and symbol maps, and opening and closing your session with the Modulation Toolkit VIs.

Palette Object	Description
<u>MT Generate</u> <u>System</u> <u>Parameters</u>	Calculates parameters for use with modulation and demodulation VIs. The system parameters cluster from this VI wires to the corresponding parameter of the appropriate demodulation or modulation VI.
MT Generate Synchronization Parameters	Constructs an FSK, PSK, QAM, MSK, PAM, ASK, or CPM synchronization parameters cluster that you can pass to a demodulation VI, using an array of either synchronization bits or synchronization numbers. The generated synchronization parameters cluster avoids phase ambiguity at the demodulator, thus allowing recovery of the actual message bits.
<u>MT Generate</u> <u>Filter</u> <u>Coefficients</u>	Calculates filter coefficients for pulse-shaping and matched filters applied by the digital modulation VIs and demodulation VIs.
<u>MT Generate</u> Symbol Map	Generates an arbitrarily shaped symbol map representing the I/Q constellation points. This polymorphic VI consists of two instances for generating either rectangular or circular I/Q symbol maps.
<u>MT Pack Bits</u>	Packs data to an unsigned or signed integer array. You can specify the order in which the bits are packed into the integer array either as MSB First or LSB First . You can also specify the integer format to be Signed or Unsigned using the integer format parameter.
<u>MT Map Bits to</u> <u>Symbols</u>	Maps bits to complex valued symbols for PSK, QAM, PAM, ASK, and CPM modulation schemes and frequency deviations for FSK and MSK modulation schemes.
MT Pulse	Generates a sampled continuous-time I/Q baseband

Shaping Filter	waveform from the discrete-valued input symbols by applying a pulse-shaping filter.
<u>MT Matched</u> <u>Filter</u>	Applies specified matched-filter coefficients to the input complex I/Q baseband waveform. The VI returns the matched filtered output with a duration equal to an integer number of symbols (thus ensuring continuability from one iteration to the next).
<u>MT Decimate</u> <u>Oversampled</u> <u>Waveform</u>	Decimates the input complex waveform and returns the decimated output complex waveform . Use this VI to decimate the matched filtered waveform at the output of the <u>MT Matched Filter</u> VI to recover the symbols corresponding to the ideal symbol timing location.
<u>MT Align To</u> Ideal Symbols	Locates the first occurrence of the ideal symbol timing instant in the matched filtered input complex waveform . It then applies phase-continuous resampling to align the first sample of the input complex waveform to the ideal symbol timing instant. The returned waveform is symbol-time aligned such that its first sample corresponds to the optimal (ideal) symbol instant.
<u>MT Map</u> Symbols to Bits	Maps complex valued PSK-, QAM-, PAM-, ASK-, FSK-, MSK-, and CPM-modulated symbols to an output bit stream based on a user-specified symbol map.
MT UnPack Bits	Unpacks an array of signed or unsigned integer data. You can specify the packed bit order as MSB First or LSB First . You can also specify the integer format to be Signed or Unsigned .
MT Calculate Worst Case Scaling Factor (Modulation and Resample)	Calculates the maximum possible peak amplitude, after pulse-shape filtering and resampling, of a waveform. This calculation is based on system parameters for a given modulation scheme.
MT Calculate Power Adjustment and PAPR	Returns the peak-to-average power ratio (PAPR) and the full-scale power adjustment.

MT Align To Ideal Symbols VI

Locates the first occurrence of the ideal symbol timing instant in the matched filtered **input complex waveform**. It then applies phase-continuous resampling to align the first sample of the **input complex waveform** to the ideal symbol timing instant. The returned waveform is symbol-time aligned such that its first sample corresponds to the optimal (ideal) symbol instant.

Ŧ

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Align to Ideal Symbols (Generic)

This polymorphic instance aligns the first sample of the **input complex waveform** to the ideal symbol timing instant for PAM, ASK, QAM, and nonoffset PSK modulation schemes.



- input complex waveform specifies the continuous-time samples of the modulated complex baseband waveform data. Wire the **output complex waveform** returned from the <u>MT Matched Filter</u> VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- matched filter coefficients specifies an ordered array containing coefficients for the matched filter that has been applied to the input complex waveform. Wire the matched filter coefficients parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. Ensure that the value of the samples per symbol parameter in this VI matches the matched samples per symbol value used to generate these filter coefficients.
- samples per symbol specifies the number of samples per information symbol in the input complex waveform. The default is 16.
- reset? specifies whether the VI continues to align the input complex waveform to ideal symbol timing instants using the previous iteration states. The VI resets on the first call and when reset? is set to TRUE. The default is TRUE.
- error in (no error) can accept error information wired from

previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- output complex waveform returns the symbol-time-aligned I/Q baseband waveform. Wire this parameter to the <u>MT Decimate</u> <u>Oversampled Waveform</u> VI followed by the <u>MT Map Symbols to Bits</u> VI to obtain the recovered bits corresponding to the modulated waveform.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **DEL** dt returns time interval between data points in the Y array. The reciprocal of this value is equal to the system sampling rate.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.

waveform alignment offset returns the location of the ideal symbol timing instant in samples relative to the beginning of the input complex waveform. This offset was applied to the output complex waveform. Multiply this value by the dt element of the input complex waveform to calculate the location of the ideal symbol timing instant in seconds.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Align To Ideal Symbols (Offset PSK)

This polymorphic instance aligns the first sample of the **input complex waveform** to the ideal symbol timing instant for an offset PSK (OPSK) modulation scheme. This instance returns an **output complex waveform (offset removed)** with the 1/2 symbol I/Q offset removed. Wire the **output complex waveform (offset removed)** to the<u>MT</u> <u>Decimate Oversampled Waveform</u> VI to obtain the demodulated soft symbol decisions.



- input complex waveform specifies the continuous-time samples of the modulated complex baseband waveform data. Wire the **output complex waveform** returned from the <u>MT Matched Filter</u> VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **matched filter coefficients** specifies an ordered array containing coefficients for the matched filter that has been applied to the **input complex waveform**. Wire the **matched filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. Ensure that the value of the **samples per symbol** parameter in this VI matches the **matched samples per symbol** value used to generate these filter coefficients.
- samples per symbol specifies the number of samples per information symbol in the input complex waveform. The default is 16.
- **reset?** specifies whether the VI continues to align the **input**

complex waveform to ideal symbol timing instants using the previous iteration states. The VI resets on the first call and when **reset?** is set to TRUE. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- output complex waveform returns the symbol-time-aligned I/Q baseband waveform. Wire this parameter to the <u>MT Decimate</u> <u>Oversampled Waveform</u> VI followed by the <u>MT Map Symbols to Bits</u> VI to obtain the recovered bits corresponding to the modulated waveform.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.

output complex waveform (offset removed) returns the matchedfiltered continuous-time I/Q baseband waveform with the 1/2 symbol I/Q timing offset removed. Wire this parameter to the <u>MT</u> <u>Decimate Oversampled Waveform</u> VI followed by <u>MT Map Symbols</u> to <u>Bits</u> VI to obtain the recovered bits corresponding to this offsetremoved symbol-time-aligned waveform.

- **ID** returns the trigger (start) time of the **Y** array.
- **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.
- Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- waveform alignment offset returns the location of the ideal symbol timing instant in samples relative to the beginning of the input complex waveform. This offset was applied to the output complex waveform. Multiply this value by the **dt** element of the input complex waveform to calculate the location of the ideal symbol timing instant in seconds.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more

information about the error.

MT Align To Ideal Symbols (FSK)

This polymorphic instance aligns the first sample of the **input phase waveform** to the ideal symbol timing instant for an FSK modulation scheme.



input phase waveform specifies the continuous-time samples of the modulated complex baseband waveform data. Wire the output complex waveform returned from the <u>MT Matched Filter</u> VI to this parameter.



Note For proper operation of this VI, the following requirements must be met:

- The sampling rate of the input phase waveform must be an even integer multiple of the expected symbol rate. Use the <u>MT Resample (Complex</u> <u>Cluster</u>) VI to adjust the input phase waveform sampling rate if necessary.
- An appropriate matched filter must be previously applied to the input phase waveform. Use the MT Matched Filter VI to apply the matched filter.
- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- **Y** specifies the matched-filtered phase of the FSK-modulated baseband waveform.
- **matched filter coefficients** specifies an ordered array containing coefficients for the matched filter that has been applied to the **input complex waveform**. Wire the **matched filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. Ensure that the value of the **samples per symbol** parameter in this VI matches the **matched samples per symbol**

value used to generate these filter coefficients.

- **samples per symbol** specifies the number of samples per information symbol in the **input complex waveform**. The default is 16.
- reset? specifies whether the VI continues to align the input complex waveform to ideal symbol timing instants using the previous iteration states. The VI resets on the first call and when reset? is set to TRUE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- output phase waveform returns the symbol-time-aligned phase of the FSK-modulated baseband waveform. Wire this parameter to the <u>MT Decimate Oversampled Waveform</u> VI followed by the <u>MT Map</u> <u>Symbols to Bits</u> VI to obtain the recovered bits corresponding to the modulated waveform.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.

Y returns the symbol-time aligned phase of the FSK-modulated waveform.

- waveform alignment offset returns the location of the ideal symbol timing instant in samples relative to the beginning of the input complex waveform. This offset was applied to the output complex waveform. Multiply this value by the dt element of the input complex waveform to calculate the location of the ideal symbol timing instant in seconds.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Align To Ideal Symbols (MSK, CPM)

This polymorphic instance aligns the first sample of the **input phase waveform** to the ideal symbol timing instant for an MSK or CPM modulation scheme.



input phase waveform specifies the continuous-time samples of the modulated complex baseband waveform data. Wire the **output complex waveform** returned from the <u>MT Matched Filter</u> VI to this parameter.



Note For proper operation of this VI, the following requirements must be met:

- The sample rate of the **input phase waveform** must be an even integer multiple of the expected symbol rate. Use the <u>MT Resample (Complex Cluster)</u> VI to adjust the **input phase waveform** sampling rate if necessary.
- An appropriate matched filter must be previously applied to the **input phase waveform**. Use the <u>MT</u> <u>Matched Filter</u> VI to apply the matched filter.
- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- **Y** specifies the matched-filtered phase of the MSK-modulated baseband waveform.
- **matched filter coefficients** specifies an ordered array containing coefficients for the matched filter that has been applied to the **input complex waveform**. Wire the **matched filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. Ensure that the value of the **samples per symbol** parameter in this VI matches the **matched samples per symbol**

value used to generate these filter coefficients.

- **samples per symbol** specifies the number of samples per information symbol in the **input complex waveform**. The default is 16.
- reset? specifies whether the VI continues to align the input complex waveform to ideal symbol timing instants using the previous iteration states. The VI resets on the first call and when reset? is set to TRUE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- output phase waveform returns the symbol-time-aligned phase of the MSK- or CPM-modulated baseband waveform. Wire this parameter to the <u>MT Decimate Oversampled Waveform</u> VI followed by the <u>MT Map Symbols to Bits</u> VI to obtain the recovered bits corresponding to the modulated waveform.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.

Y returns the symbol-time aligned phase of the MSK-modulated waveform.

- waveform alignment offset returns the location of the ideal symbol timing instant in samples relative to the beginning of the input complex waveform. This offset was applied to the output complex waveform. Multiply this value by the dt element of the input complex waveform to calculate the location of the ideal symbol timing instant in seconds.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details



Note For proper operation of this VI, the following requirements must be met:

- The sample rate of the **input complex waveform** or **input phase waveform** must be an even integer multiple of the expected symbol rate. Use the <u>MT Fractional Resample VI</u> to adjust the sampling rate of the **input complex waveform** or **input phase waveform** if necessary.
- An appropriate matched filter must be previously applied to the **input complex waveform** or **input phase waveform** using the <u>MT Matched Filter</u> VI.

MT Matched Filter VI

Applies specified matched-filter coefficients to the input complex I/Q baseband waveform. The VI returns the matched filtered output with a duration equal to an integer number of symbols (thus ensuring continuability from one iteration to the next).

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Apply Matched Filter (Generic)

This polymorphic instance applies a matched filter to a PAM, ASK, QAM, and PSK-modulated input waveform.



input complex waveform specifies the continuous-time modulated complex baseband waveform data.

- Note The sampling rate of the input complex waveform must be an even integer multiple of the samples per symbol value. Use the <u>MT Resample (Complex Cluster)</u> VI to adjust the sampling rate before matched filter application.
- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- matched filter coefficients specifies an ordered array containing sample-spaced coefficients for the desired matched filter. Wire the matched filter coefficients parameter of the <u>MT Generate Filter</u> <u>Coefficients</u> VI to this parameter. Ensure that the value of samples per symbol matches the matched samples per symbol value used to generate the filter coefficients.
- samples per symbol specifies the number of samples per information symbol in the input complex waveform. The default is 16.
- **reset?** specifies whether the VI continues matched filtering using the previous iteration states. The matched filtering process is reset on the first call and when **reset?** is set to TRUE. When **reset?** is set to FALSE, matched filtering begins from the last iteration state

on subsequent calls. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the output matched filtered continuous-time I/Q baseband waveform.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Apply Matched Filter (FSK)

This polymorphic instance applies a matched filter to an input FSKmodulated waveform. The VI first performs phase differentiation, followed by matched filtering on the phase of the **input complex waveform**.



input complex waveform specifies the continuous-time modulated complex baseband waveform data.

- Note The sampling rate of the input complex waveform must be an even integer multiple of the samples per symbol value. Use the <u>MT Resample (Complex Cluster)</u> VI to adjust the sampling rate before matched filter application.
- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- matched filter coefficients specifies an ordered array containing sample-spaced coefficients for the desired matched filter. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. Ensure that the value of samples per symbol matches the matched samples per symbol value used to generate the filter coefficients.
- samples per symbol specifies the number of samples per information symbol in the input complex waveform. The default is 16.
- **symbol phase continuity** specifies whether the phase transitions between symbols are continuous.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, discontinuous phase FSK (DPFSK).

- **reset?** specifies whether the VI continues matched filtering using the previous iteration states. The matched filtering process is reset on the first call and when **reset?** is set to TRUE. When **reset?** is set to FALSE, matched filtering begins from the last iteration state on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output phase waveform** returns the output matched filtered continuous-time phase waveform.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **besilt** returns the time between values in the **Y** array.
 - **Y** returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this

complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
MT Apply Matched Filter (MSK)

This polymorphic instance applies a matched filter to an input MSKmodulated waveform. The VI first performs phase differentiation, followed by matched filtering on the phase of the **input complex waveform**.



input complex waveform specifies the continuous-time modulated complex baseband waveform data.

- Note The sampling rate of the input complex waveform must be an even integer multiple of the samples per symbol value. Use the <u>MT Resample (Complex Cluster)</u> VI to adjust the sampling rate before matched filter application.
- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>1</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- matched filter coefficients specifies an ordered array containing sample-spaced coefficients for the desired matched filter. Wire the matched filter coefficients parameter of the MT Generate Filter Coefficients VI to this parameter. Ensure that the value of samples per symbol matches the matched samples per symbol value used to generate the filter coefficients.
- samples per symbol specifies the number of samples per information symbol in the input complex waveform. The default is 16.
- **reset?** specifies whether the VI continues matched filtering using the previous iteration states. The matched filtering process is reset on the first call and when **reset?** is set to TRUE. When **reset?** is

set to FALSE, matched filtering begins from the last iteration state on subsequent calls. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output phase waveform** returns the output matched filtered continuous-time phase waveform.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **besize the set whet a set whether a set whether a set of the between the set of the between the set of the between the set of the s**
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Apply Matched Filter (CPM)

This polymorphic instance applies a matched filter to a specified CPMmodulated waveform. The VI first performs phase differentiation, followed by matched filtering on the phase of the input complex waveform.



Note Matched filter coefficients can be generated using the <u>MT</u> <u>Generate Filter Coefficients</u> VI or by specifying custom matched filter coefficients. The VI does not verify absence of intersymbol interference in custom filter coefficients before applying the matched filter.



input complex waveform specifies the continuous-time modulated complex baseband waveform data.



Note The sampling rate of the input complex waveform must be an even integer multiple of the samples per symbol value. Use the <u>MT Resample (Complex Cluster</u>) VI to adjust the sampling rate before matched filter application.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- matched filter coefficients specifies an ordered array containing sample-spaced coefficients for the desired matched filter. Wire the matched filter coefficients parameter of the <u>MT Generate Filter</u> <u>Coefficients</u> VI to this parameter. Ensure that the value of samples per symbol matches the matched samples per symbol value used to generate the filter coefficients.
- **samples per symbol** specifies the number of samples per

information symbol in the **input complex waveform**. The default is 16.

- **reset? (T)** specifies whether the VI continues matched filtering using the previous iteration states. The matched filtering process is reset on the first call and when **reset?** is set to TRUE. When **reset?** is set to FALSE, matched filtering begins from the last iteration state on subsequent calls.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output phase waveform** returns the output matched filtered continuous-time phase waveform.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **besilon dt** returns the time between values in the **Y** array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.

error out passes error or warning information out of a VI to be used

by other VIs. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

In the absence of channel impairments, the first sample of the matchedfiltered output for every iteration corresponds to an ideal symbol timing instant. In the presence of channel impairments, use the <u>MT Align To</u> <u>Ideal Symbols</u> VI to locate and realign the matched-filtered waveform at the ideal symbol timing instant.



Note Generate matched-filter coefficients using the <u>MT Generate</u> <u>Filter Coefficients</u> VI or by specifying custom matched filter coefficients. The VI does not verify absence of intersymbol interference in custom filter coefficients before applying the matched filter.

MT Pulse Shaping Filter VI

Generates a sampled continuous-time I/Q baseband waveform from the discrete-valued input symbols by applying a pulse-shaping filter.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Apply Pulse Shaping Filter (Generic)

This polymorphic instance applies a pulse-shaping filter to an input stream comprised of PAM, ASK, QAM, and nonoffset PSK-modulated symbols. Separate polymorphic instances are available for offset PSK, FSK, and MSK.



- **symbols** specifies an array of symbols (comprised of frequency deviations) corresponding to information bits. Wire the **symbols** parameter of the <u>MT Map Bits to Symbols</u> VI to this parameter.
- **pulse shaping filter coefficients** specifies an ordered array containing sample-spaced coefficients for the desired pulseshaping filter. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. Ensure that the value of the **samples per symbol** passed to this VI matches the **pulse shaping samples per symbol** value used to generate the filter coefficients.
 - Note When reset? is set to TRUE, there is a transient response of ½ the filter length at the start of the modulated signal, and the returned data is shortened by approximately ½ the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- samples per symbol specifies the number of samples per information symbol in the output complex waveform. Multiply this value by the symbol rate to determine the system sampling rate. The default is 16.
- **symbol rate** specifies the symbol rate in symbols per second. The default is 1.0.
- **reset?** specifies whether the VI continues pulse-shape filtering using the previous iteration states. The pulse-shaping process is reset on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE to begin pulse shaping at the previous iteration state on

subsequent calls. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the output pulse-shaped continuous-time I/Q baseband waveform.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Apply Pulse Shaping Filter (Offset PSK)

This polymorphic instance applies a pulse-shaping filter to an input stream comprised of offset PSK-modulated symbols. The VI shifts the Q-phase baseband data by $\frac{1}{2}$ symbol relative to the I-phase baseband data, so the first Q-symbol lasts for the duration of 1.5 symbol periods.



Note Use the generic instance of this VI for nonoffset PSK-modulated signals.



- **symbols** specifies an array of symbols (comprised of frequency deviations) corresponding to information bits. Wire the **symbols** parameter of the <u>MT Map Bits to Symbols</u> VI to this parameter.
- **pulse shaping filter coefficients** specifies an ordered array containing sample-spaced coefficients for the desired pulseshaping filter. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. Ensure that the value of the **samples per symbol** passed to this VI matches the **pulse shaping samples per symbol** value used to generate the filter coefficients.
 - Note When reset? is set to TRUE, there is a transient response of ½ the filter length at the start of the modulated signal, and the returned data is shortened by approximately ½ the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- samples per symbol specifies the number of samples per information symbol in the output complex waveform. Multiply this value by the symbol rate to determine the system sampling rate. The default is 16.
- **symbol rate** specifies the symbol rate in symbols per second. The default is 1.0.
- **reset?** specifies whether the VI continues pulse-shape filtering using the previous iteration states. The pulse-shaping process is

reset on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE to begin pulse shaping at the previous iteration state on subsequent calls. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the output pulse-shaped continuous-time I/Q baseband waveform.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu

for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Apply Pulse Shaping Filter (FSK)

This polymorphic instance applies a pulse-shaping filter to an input stream comprised of FSK-modulated symbols. The VI performs filtering using the specified **pulse shaping filter coefficients** followed by accumulation on the phase of the information symbols to generate the oversampled FSK-modulated complex baseband waveform. This process is the digital equivalent of continuous-time analog frequency modulation.

Note With symbol phase continuity set to discontinuous, the VI ignores the pulse shaping filter coefficients parameter. Instead, it generates the baseband waveform after sampling the output of *M* separate complex exponential oscillators.



symbols specifies an array of symbols (comprised of frequency deviations) corresponding to information bits. Wire the **symbols** parameter of the <u>MT Map Bits to Symbols</u> VI to this parameter.

- **pulse shaping filter coefficients** specifies an ordered array containing sample-spaced coefficients for the desired pulseshaping filter. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. Ensure that the value of the **samples per symbol** passed to this VI matches the **pulse shaping samples per symbol** value used to generate the filter coefficients.
 - Note When reset? is set to TRUE, there is a transient response of ½ the filter length at the start of the modulated signal, and the returned data is shortened by approximately ½ the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.

samples per symbol specifies the number of samples per information symbol in the output complex waveform. Multiply this value by the symbol rate to determine the system sampling rate. The default is 16.

- **symbol rate** specifies the symbol rate in symbols per second. The default is 1.0.
- **symbol phase continuity** specifies whether the phase transitions between symbols are continuous.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, discontinuous phase FSK (DPFSK).

reset? specifies whether the VI continues pulse-shape filtering using the previous iteration states. The pulse-shaping process is reset on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE to begin pulse shaping at the previous iteration state on subsequent calls. The default is TRUE.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output phase waveform** returns the output pulse-shaped continuous-time phase waveform.

- **ID** returns the trigger (start) time of the **Y** array.
- **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.
- **Y** returns the complex-valued signal-only baseband phase waveform.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Apply Pulse Shaping Filter (MSK, CPM)

This polymorphic instance applies a pulse-shaping filter to an input stream comprised of MSK- or CPM-modulated symbols. The VI performs filtering using the specified **pulse shaping filter coefficients** followed by accumulation on the phase of the information symbols to generate the oversampled MSK- or CPM-modulated complex baseband waveform. This process is the digital equivalent of continuous-time analog frequency modulation.



- **symbols** specifies an array of symbols (comprised of frequency deviations) corresponding to information bits. Wire the **symbols** parameter of the <u>MT Map Bits to Symbols</u> VI to this parameter.
- **pulse shaping filter coefficients** specifies an ordered array containing sample-spaced coefficients for the desired pulseshaping filter. Wire the **pulse shaping filter coefficients** parameter of the <u>MT Generate Filter Coefficients</u> VI to this parameter. Ensure that the value of the **samples per symbol** passed to this VI matches the **pulse shaping samples per symbol** value used to generate the filter coefficients.
 - Note When reset? is set to TRUE, there is a transient response of ½ the filter length at the start of the modulated signal, and the returned data is shortened by approximately ½ the filter length. When reset? is set to FALSE, the VI uses data from the previous iteration to eliminate the transient.
- samples per symbol specifies the number of samples per information symbol in the output complex waveform. Multiply this value by the symbol rate to determine the system sampling rate. The default is 16.
- **symbol rate** specifies the symbol rate in symbols per second. The default is 1.0.
- **reset?** specifies whether the VI continues pulse-shape filtering

using the previous iteration states. The pulse-shaping process is reset on the first call and when **reset?** is set to TRUE. Set **reset?** to FALSE to begin pulse shaping at the previous iteration state on subsequent calls. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the output pulse-shaped continuous-time I/Q baseband waveform.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **dt** returns time interval between data points in the **Y** array. The reciprocal of this value is equal to the system sampling rate.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and

select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

This VI first upsamples (zero-insertion) the **symbols** array by a factor equal to the specified **samples per symbol** value, then filters the symbols stream with the user-specified pulse-shaping filter.



Note Pulse-shaping filter coefficients can be generated using the <u>MT Generate Filter Coefficients</u> VI or by specifying custom pulse-shaping filter coefficients. The VI does not verify absence of intersymbol interference in custom filter coefficients before applying the pulse-shaping filter.

MT Calculate Power Adjustment and PAPR VI

Returns the peak-to-average power ratio (**PAPR**) and the full-scale power adjustment.

In the NI-RFSG peak-power mode, after scaling the waveform to between +1 and -1, the waveform power may not be at the desired level. You can use the **full scale adjustment** parameter to compensate for the loss in power and the **PAPR** parameter to change the average power of the modulated waveform to the peak power specifically required for streaming mode.

Note The input complex waveform must be scaled by the worst case scaling factor generated by the <u>MT Calculate Worst Case</u> Scaling Factor (Modulation and Resample) VI.



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- waveform scaling factor specifies the scaling factor used on the waveform to position it between +1 and -1. Typically, this parameter would be wired from the waveform scaling factor parameter of the MT Calculate Worst Case Scaling Factor (Modulation and Resample) VI.

This VI uses this parameter to scale the input complex waveform.

reset? specifies whether to calculate the **full scale adjustment** and **PAPR** parameters. On first call, **reset?** is set to TRUE by default. This parameter is TRUE by default for subsequent iterations.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **full scale adjustment** returns the number by which the NI-RFSG power level must be adjusted.
- PAPR returns the peak-to-average power ratio of the scaled input complex waveform, where the scaling factor is derived from the MT Calculate Worst Case Scaling Factor (Modulation and Resample) VI.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Calculate Worst Case Scaling Factor (Modulation and Resample) VI

Calculates the maximum possible peak amplitude, after pulse-shape filtering and resampling, of a waveform. This calculation is based on system parameters for a given modulation scheme.

By using this VI with the NI-RFSG instrument driver, you can ensure that your waveform lies between +1 and -1, which is a requirement for the NI-RFSG peak-power generation mode.

For the frequency modulation schemes (FSK, MSK, and CPM), there is no known analytical method to calculate the maximum amplitude that accounts for effects such as resampling, filtering, and so on. Empirically, a **waveform scaling factor** of 0.8 should be sufficient for most of your applications. This value results in a maximum dynamic range loss of about 2 dB on the RF signal generator.

If you receive errors related to this value, your signal might be aliased. For more information about aliasing, refer to the <u>MT Modulate CPM</u> or <u>MT Modulate FSK</u> VI.

▼

Use the pull-down menu to select an instance of this VI.

Select an instance

MT Calculate Scaling Factor (ASK)

Calculates the maximum possible waveform amplitude for the ASK modulation scheme.

ASK System Parameters 01011101 waveform scaling factor pulse shaping filter coeffi... error in (no error)

- ASK system parameters specifies parameter values defining the ASK system. Wire this parameter to the corresponding system parameters cluster of the MT Modulate ASK, MT Demodulate ASK, or MT Detect ASK VIs.
 - **samples per symbol** specifies the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** value must be an even number. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to the desired deviation frequency. The number of ASK levels in the array must be 2^N, where N is the number of bits per symbol. The vector length for the symbol(s) farthest from the origin must be 1.
- **pulse shaping filter coefficients** specifies an ordered array of filter coefficients corresponding to the desired filter response for the pulse-shaping filter used in modulation. The number of coefficients depends on the pulse-shaping filter characteristics specified in the pulse-shaping filter. Refer to the <u>MT Generate Filter Coefficients</u> VI for more information about pulse-shaping filter coefficients.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more

information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- waveform scaling factor returns the inverse of the maximum amplitude of the I/Q waveform based on the specified system parameters.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Calculate Scaling Factor (PAM)

Calculates the maximum possible waveform amplitude for the PAM modulation scheme.

PAM System Parameters 0101101 waveform scaling factor pulse shaping filter coeffi... error in (no error)

- **PAM system parameters** specifies parameter values defining the PAM system. Wire this parameter to the corresponding **system parameters** cluster of the <u>MT Modulate PAM</u>, <u>MT Demodulate</u> <u>PAM</u>, or <u>MT Detect PAM</u> VIs.
 - **samples per symbol** specifies the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** value must be an even number. The default is 16.
 - **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of PAM levels in the array is 2^N, where N is the number of bits per symbol. The vector length for the symbol(s) farthest from the origin is 1.
- **pulse shaping filter coefficients** specifies an ordered array of filter coefficients corresponding to the desired filter response for the pulse-shaping filter used in modulation. The number of coefficients depends on the pulse-shaping filter characteristics specified in the pulse-shaping filter. Refer to the <u>MT Generate Filter Coefficients</u> VI for more information about pulse-shaping filter coefficients.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- waveform scaling factor returns the inverse of the maximum amplitude of the I/Q waveform based on the specified system parameters.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Calculate Scaling Factor (QAM)

Calculates the maximum possible waveform amplitude for the QAM modulation scheme.

QAM System Parameters 01011101 waveform scaling factor pulse shaping filter coeffi... error in (no error)

- **QAM system parameters** specifies parameter values defining the QAM system. Wire this parameter to the corresponding **system parameters** cluster of the <u>MT Modulate QAM</u>, <u>MT Demodulate</u> <u>QAM</u>, or <u>MT Detect QAM</u> VIs.
 - **samples per symbol** specifies the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** value must be an even number. The default is 16.
 - **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^N, where N is the number of bits per symbol. The length of the vector for the symbol(s) farthest from the origin must be 1.
- **pulse shaping filter coefficients** specifies an ordered array of filter coefficients corresponding to the desired filter response for the pulse-shaping filter used in modulation. The number of coefficients depends on the pulse-shaping filter characteristics specified in the pulse-shaping filter. Refer to the MT Generate Filter Coefficients VI for more information about pulse-shaping filter coefficients.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more

information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- waveform scaling factor returns the inverse of the maximum amplitude of the I/Q waveform based on the specified system parameters.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Calculate Scaling Factor (PSK)

Calculates the maximum possible waveform amplitude for the PSK modulation scheme.



- **PSK system parameters** specifies parameter values defining the PSK system. Wire this parameter to the corresponding **system parameters** cluster of the <u>MT Modulate PSK</u>, <u>MT Demodulate PSK</u>, or <u>MT Detect PSK</u> VIs.
 - **samples per symbol** specifies the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** value must be an even number. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.
 - **differential PSK** specifies the status of the differential PSK.

disable (0)	Does not allow bit sequence encoding.
enable (1)	Allows bit sequence encoding.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.

shifted Rotates the constellation by π/M each symbol.

offset	Sets the modulation type to offset quadrature phase-
	shift keying (OQPSK). This modulation scheme is a
	form of phase-shift keying in which four different
	phase angles are used. This scheme is sometimes
	referred to as staggered quadrature phase-shift
	keying (SQPSK). For offset PSK, the ideal symbol
	timing for Q is offset by 1/2 of a symbol period from
	the ideal symbol timing for I. offset is currently only
	supported for $M = 4$.

- **pulse shaping filter coefficients** specifies an ordered array of filter coefficients corresponding to the desired filter response for the pulse-shaping filter used in modulation. The number of coefficients depends on the pulse-shaping filter characteristics specified in the pulse-shaping filter. Refer to the <u>MT Generate Filter Coefficients</u> VI for more information about pulse-shaping filter coefficients.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- waveform scaling factor returns the inverse of the maximum amplitude of the I/Q waveform based on the specified system parameters.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for

more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Close Session VI

Closes any open modulation session.

Call this VI only under the following conditions:

- An empty modulation session has been created using the <u>MT</u> <u>Open Session</u> VI.
- All attributes have been written to (set) or read from (get) the property node located in the **All Functions**»**Application Control** palette.
- The appropriate modulation VI(s) (for example, the <u>MT Reed</u> <u>Solomon Symbol Encode</u> VI) has been called with the modulation session refnum and has completed execution.



- modulation session specifies the modulation session refnum, which you can use to configure the behavior and operation of the appropriate modulation VI that accepts the modulation session refnum at its input. Use the <u>MT Open Session</u> VI to create the modulation session refnum. The default is empty.
 - Note You can navigate to the All Functions»Application Control palette and drop a property node on the block diagram. After you wire the modulation session refnum to the input of the property node, all the modulation-specific attributes are accessible.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
MT Decimate Oversampled Waveform VI

Decimates the **input complex waveform** and returns the decimated **output complex waveform**. Use this VI to decimate the matched filtered waveform at the output of the <u>MT Matched Filter</u> VI to recover the symbols corresponding to the ideal symbol timing location.

<u>Details</u>



input complex waveform specifies the modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- **decimation factor** specifies the extent to which the **input complex waveform** is decimated. This parameter is ignored on subsequent calls when **reset?** is set to FALSE.
 - Tip If you are recovering the ideal symbol timing instant, ensure that this input value matches the samples per symbol parameter to the <u>MT Matched Filter</u> VI.
- offset specifies the starting index of the Y array within the input complex waveform cluster from which decimation is performed. This parameter is ignored on subsequent calls when reset? is set to FALSE. Samples prior to the specified offset are discarded. Use this parameter to specify the first sample containing an ideal symbol timing instant. The default is 0 (no samples discarded).
- **reset?** specifies whether stored state information is cleared at subsequent calls to this VI. When **reset?** is set to TRUE,

decimation begins at the starting point specified by **offset**. When **reset?** is set to FALSE, the specified **offset** and **decimation factor** values are used on first call only, and computed on subsequent calls to ensure continuability from one iteration to the next. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the decimated I/Q baseband waveform sampled at a rate 1/(**decimation factor × dt**) seconds/sample.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **besilt of the time interval between data points in the Y array.**
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and

select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

- Note Although the <u>MT Resample (Complex Cluster</u>) VI can be used to downsample a complex waveform, the MT Decimate Oversampled Waveform VI is sometimes a more suitable choice because it does not apply filtering during the decimation process. Compared to the MT Fractional Resample VI, this VI has the following characteristics:
 - Does not introduce any additional phase shift (other than any specified nonzero **offset**)
 - Does not introduce transient effects associated with filter application
 - More computational efficiency and speed

MT Generate Symbol Map VI

Generates an arbitrarily shaped symbol map representing the I/Q constellation points. This polymorphic VI consists of two instances for generating either rectangular or circular I/Q symbol maps.

The generated symbol maps should be bundled onto the **system parameters** cluster prior to being passed to the corresponding modulation VI. You can use the rectangular instance of this VI to create a custom symbol map for a generic amplitude modulation scheme such as quadrature amplitude modulation (QAM). Use the circular instance of this VI to create a custom symbol map for a combined amplitude-phase modulation scheme.

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Generate Symbol Map (Rectangular)

This polymorphic instance generates a symbol map with a user-specified input matrix of symbol values and I/Q symbol amplitude arrays. Use the **symbol map** parameter inside the **system parameters** cluster to configure the operation of the demodulation and modulation VIs.

The **symbol values** matrix defines how the *M*-ary binary data are mapped to the discrete I/Q symbols on the constellation diagram. The optional **gain** parameter can be used to multiply the generated symbol map by a scalar value. The VI internally reads the **symbol values** matrix row-wise and uses the I/Q symbol amplitudes array values to assign the symbol amplitudes to each element in the row.



Note The number of rows in the symbol amplitudes matrix should exactly equal the length of the symbol amplitudes (Q) array.
Additionally, the number of columns of the symbol amplitudes matrix should exactly equal the length of the symbol amplitudes (I) array. If these conditions are violated, the VI returns an error.



symbol values specifies a matrix of integers that define how the *M*ary (bits per symbol) binary data are mapped to discrete I/Q symbols on the constellation diagram. Use negative values to specify holes (missing points) in the constellation diagram. The default is empty.



Note The binary representation of the symbol values is performed using the LSB first convention. For example, in 8-QAM, the symbol values matrix can be specified as

$$\begin{bmatrix} 0 & 1 & 2 \\ 3 & -1 & 4 \end{bmatrix}$$

5 6 7

This matrix corresponds to the following bit values:

- 000 100 010 110 XXX 001
- 101 011 111

The term *XXX* here refers to a hole (missing point) on the constellation plot.

symbol amplitudes (I) specifies the in-phase amplitudes of the I/Q constellation points corresponding to the **symbol values** array.

For example, in the case of 8-QAM, assume that the symbol values

 $\begin{bmatrix} 0 & 1 & 2 \\ 3 & -1 & 4 \end{bmatrix}$

are specified as $\begin{bmatrix} 5 & 6 & 7 \end{bmatrix}$, while the I-symbol amplitudes are specified as [-1 0 1]. Then, the symbol values [0 3 5] are assigned an I-amplitude of -1, the symbol values [1 6] are assigned an Iamplitude of 0, and the symbol values [2 4 7] are assigned an Iamplitude of 1. The default is empty.

symbol amplitudes (Q) specifies the quadrature-phase amplitudes of the I/Q constellation points corresponding to the **symbol values** array.

For example, in the case of 8-QAM, assume that the symbol values

 $\begin{bmatrix} 0 & 1 & 2 \\ 3 & -1 & 4 \end{bmatrix}$

are specified as $\begin{bmatrix} 5 & 6 & 7 \end{bmatrix}$, while the Q-symbol amplitudes are specified as $\begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$. Then, the symbol values $\begin{bmatrix} 0 & 1 & 2 \end{bmatrix}$ are assigned a Q-amplitude of 1, the symbol values $\begin{bmatrix} 3 & 4 \end{bmatrix}$ are assigned a Qamplitude of 0, and the symbol values $\begin{bmatrix} 5 & 6 & 7 \end{bmatrix}$ are assigned a Qamplitude of -1. The default is empty.

- **gain** specifies an optional scalar constant by which to multiply the symbol map (amplitudes defined by the I/Q symbol amplitude arrays). The default is 1.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain** **Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbol map** returns a symbol map as an ordered array to be bundled with the **system parameters** cluster that is passed to the digital demodulation and modulation VIs.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate Symbol Map (Circular)

This polymorphic instance generates a symbol map, provided a userspecified matrix of **symbol values**, a matrix representing the symbol phases in degrees, and an array representing the radii of each discrete circle. Use the **symbol map** parameter inside the **system parameters** cluster to configure the operation of the demodulation and modulation VIs. The **symbol values** matrix defines how the *M*-ary binary data are mapped to the discrete I/Q symbols on the constellation diagram. The VI maps the specified symbol values to I/Q constellation points using the polar representation $j\pi \frac{\theta_i}{180}$. The VI internally reads the **symbol**

values/symbol phases matrices row-wise. Each row of symbol values/symbol phases is then mapped onto I/Q symbol points, which are located on a circle with a radius that is specified in the **symbol radii** array.

 $\overline{\mathbb{N}}$

Note The dimensions of the symbol amplitudes matrix should exactly match the dimensions of the **symbol phases** matrix. Additionally, the number of rows of each matrix should equal the length of the **symbol radii** array. If these conditions are violated, the VI returns an error.



symbol values specifies a matrix of integers that define how the *M*ary (bits per symbol) binary data are mapped to discrete I/Q symbols on the constellation diagram. Use negative values to specify holes (missing points) in the constellation diagram. The default is empty.



Note The binary representation of the symbol values is performed using the LSB first convention. For example, in 8-QAM, the symbol values matrix can be specified as

```
\begin{bmatrix} 0 & 1 & 2 \\ 3 & -1 & 4 \\ 5 & 6 & 7 \end{bmatrix}
```

This matrix corresponds to the following bit values:

 000
 100
 010

 110
 XXX
 001

 101
 011
 111

The term *XXX* here refers to a hole (missing point) on the constellation plot.

- **symbol phases** specifies the phases θ (in degrees) corresponding to the location of the elements of the symbol values matrix on the I/Q constellation diagram. The default is empty.
- **symbol radii** specifies the radii of discrete concentric circles that contain the I/Q constellation points. There is a one-to-one mapping between every entry on the **symbol radii** array and every row in the **symbol values/symbol phases (degrees)** matrix. The default is empty.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbol map** returns a symbol map as an ordered array to be bundled with the **system parameters** cluster that is passed to the digital demodulation and modulation VIs.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The following example demonstrates 8-QAM symbol map creation.

```
\begin{bmatrix} 0 & 1 & 2 \\ 3 & -1 & 4 \\ 5 & 6 & 7 \end{bmatrix}
Define the I symbol amplitude array as \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}
Define the Q symbol amplitude array as \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}
Define gain as 1.0
```

These settings result in the following symbol map.



MT Generate Filter Coefficients VI

Calculates filter coefficients for pulse-shaping and matched filters applied by the digital modulation VIs and demodulation VIs.

Details



pulse shaping filter (none) specifies the class of filter to generate. selection here must correspond with the **modulation type** parameter setting as shown in the following table. The default is **none**.

Modulation Type		Valid Pulse-Shaping Type
PSK, QAM, ASK, PAM	none (0)	Note When you select none, the VI ignores the filter length parameter an generates a filter that is always one symbol long.
	Raised Cosine (1)	Pulse Shaping Filter 1.0 800.0 m 600.0 m 400.0 m 200.0 m 0.0 -200.0 m 0.1 2 3 4 5 6 7 8 Symbols











pulse shaping samples per symbol specifies the number of desire samples per symbol for the pulse-shaping filter. If the pulse-shaping is used for demodulation, this parameter value must match the **samp per symbol** element of the system parameters cluster passed to the demodulation VI. Specify an even number greater than 2. The defaul 16.

- Note If the pulse-shaping and matched filter coefficients are ι by the same demodulation or modulation VI, the **pulse shapin samples per symbol** and **matched samples per symbol** valu must be identical.
- matched samples per symbol specifies the number of desired sam per symbol for the demodulation matched filter. This parameter value must match the samples per symbol element of the system parame cluster passed to the digital demodulation VI. Specify an even numbe greater than 2. The default is 16.
- **filter parameter** specifies either alpha (rolloff for raised cosine and square root raised cosine filters), or *BT* (the product of the -3 dB bandwidth and the symbol period for a Gaussian filter). This paramet ignored when the **pulse shaping filter** parameter is set to **none**. The default is 0.5.
- **modulation type** specifies the type of modulation or demodulation for which to generate filter coefficients. The default is FSK.

FSK	(0)	Generates FSK filter coefficients.
-----	-----	------------------------------------

- **PSK** (1) Generates PSK filter coefficients.
- **QAM** (2) Generates QAM filter coefficients.

MSK (3) Generates MSK filter coefficients.

PAM (4) Generates PAM filter coefficients.

ASK (5) Generates ASK filter coefficients.

CPM (6) Generates CPM filter coefficients.

- **filter length** specifies the desired length of the pulse-shaping filter, ir symbols. The length of the matched filter can vary according to filter and modulation type. Valid values are > 1. The default is 8.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should b bypassed in the event of errors from other VIs. Right-click the front p error in control and select Explain Error or Explain Warning from t shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE (checkmark) 1

indicate a warning or that no error occurred. Right-click the from panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front performer in control and select **Explain Error** or **Explain Warning** f the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click t front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **pulse shaping filter coefficients** returns an ordered array of filter coefficients corresponding to the desired filter response for the pulse shaping filter used in modulation. The number of coefficients depend the pulse-shaping filter characteristics specified in the **pulse shaping** filter parameter.

Number of Coefficients	Size for FSK	Size for MSK	Size for PSK/QAM/ASK/I
none	(samples/symbol)	(samples/symbol)	(samples/symbol,
raised cosine	(samples/symbol) × filter length + 1	(samples/symbol) × filter length + 1	(samples/symbol) filter length + 1
root raised cosine	(samples/symbol) × filter length + 1	(samples/symbol) × filter length + 1	(samples/symbol) filter length + 1
Gaussian	(samples/symbol) × filter length + 1	(samples/symbol) × filter length + 1	N/A

matched filter coefficients returns an ordered array of filter coefficients corresponding to the desired filter response for the matched filter use demodulation.

Number of Coefficients	Size for FSK	Size for MSK	Size for PSK/QAM/ASK/I
none	(samples/symbol)	(samples/symbol)	(samples/symbol
raised cosine	1	1	1
root raised	(samples/symbol)	(samples/symbol)	(samples/symbol)

cosine	× filter length + 1	× filter length + 1	filter length + 1
Gaussian	14 × (samples/symbol) + 1	3 × (samples/symbol) – 1	N/A

- error out passes error or warning information out of a VI to be used other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Status** is TRUE (X) if an error occurred or FALSE (checkmark) 1 indicate a warning or that no error occurred. Right-click the from panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** fror the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click t front panel **error out** indicator and select **Explain Error** or **Expl Warning** from the shortcut menu for more information about the error.

Details

The calculated filter coefficients are used during <u>modulation</u> to reduce the bandwidth of the transmitted signal and during <u>demodulation</u> to reduce intersymbol interference. The pulse-shaping filter can be used either in transmission or for demodulation of <u>PSK-</u>, <u>FSK-</u>, <u>MSK-</u>, <u>ASK-</u>, PAM-, <u>QAM-</u>, and <u>CPM-</u>modulated signals. The matched filter is only used for demodulation.

MT Generate Synchronization Parameters VI

Constructs an FSK, PSK, QAM, MSK, PAM, ASK, or CPM **synchronization parameters** cluster that you can pass to a demodulation VI, using an array of either synchronization bits or synchronization numbers. The generated **synchronization parameters** cluster avoids phase ambiguity at the demodulator, thus allowing recovery of the actual message bits.

If the **synchronization parameter** passed to the demodulation VI is left empty, the constellation of the demodulated waveform has a 90° carrier phase ambiguity for two-dimensional modulation schemes (QAM, QPSK and higher order PSK) and a 180° carrier phase ambiguity for onedimensional modulation schemes (PAM/ASK/BPSK).

Use the pull-down menu to select an instance of this VI.

Select an instance

•

MT Generate FSK Synchronization Parameters (bit array)

This polymorphic instance generates synchronization parameters for FSK demodulation using a synchronization sequence directly represented by the **sync bits** input array.

FSK system parameters 🚥	0101101	FSK synchronization parameters
sync bits 🚽 error in (no error) 😅		error out
expected sync location (sym		
sync location uncertainty (sync indent (symbols)		

FSK system parameters specifies parameter values defining the FSK system. Wire the **FSK system parameters** cluster returned by the FSK (M) or FSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- $\overline{\mathbb{N}}$
 - Note If the symbol phase continuity element of the FSK system parameters cluster is set to discontinuous, no pulse-shaping filter can be applied, and the matched filter coefficients parameter is ignored. Refer to the FSK (M) or FSK (Map) instance of the MT Generate System Parameters VI for more information about these parameters.
- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of FSK levels in the array must be 2^N, where N is the number of bits per symbol.
- **symbol phase continuity** specifies whether the <u>phase</u> transitions between symbols are continuous.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, discontinuous phase FSK (DPFSK).

sync bits specifies an array of synchronization bits used to generate the **FSK synchronization parameter** cluster. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can wire a 16-bit array with the above entries to the **sync bits** parameter. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.

- Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.
- **expected sync location** specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1. The default is 10.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front

panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **FSK synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the sync sequence generated as a result of mapping the sync bits/numbers to symbols using the FSK symbol map. For FSK, the real portion of this array represents the frequency deviation of the symbol used to synchronize the bit stream.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.
 - **EVALUATE:** sync indent returns the specified sync indent.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - source describes the origin of the error or warning. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

MT Generate FSK Synchronization Parameters (number array)

This polymorphic instance generates synchronization parameters for FSK demodulation using a synchronization sequence defined by a **sync numbers** array in which each number represents a group of bits.



FSK system parameters specifies parameter values defining the FSK system. Wire the **FSK system parameters** cluster returned by the FSK (M) or FSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- Note If the symbol phase continuity element of the FSK system parameters cluster is set to discontinuous, no pulse-shaping filter can be applied, and the matched filter coefficients parameter is ignored. Refer to the FSK (M) or FSK (Map) instance of the MT Generate System Parameters VI for more information about these parameters.
- samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of FSK levels in the array must be 2^{*N*}, where *N* is the number of bits per symbol.
- **symbol phase continuity** specifies whether the <u>phase</u> transitions between symbols are continuous.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, discontinuous

- **sync numbers** specifies an array of synchronization numbers in which the binary representation of each number is used to generate the sync bit array. By default, the internal conversion of each sync number into the corresponding synchronization bit sequence is performed with the most significant bit first (left-aligned) approach. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can specify **sync numbers** as an array with a single element xFFFE. Specify the **bits per sync number** value in this case to be 16. You can also specify **sync numbers** as an array of two elements (xFF and xFE), and specify the **bits per sync number** value as 8. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.
 - $\overline{\mathbb{N}}$

Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.

- **bits per sync number** specifies the number of bits corresponding to each sync number in the **sync numbers** array. The default is 32.
- **expected sync location** specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- sync location uncertainty specifies the number of symbols before or after the expected sync location where the first symbol of the sync sequence may be located. This parameter is ignored if the expected sync location parameter is set to -1. The default is 10.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **FSK synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the sync sequence generated as a result of mapping the sync bits/numbers to symbols using the FSK symbol map. For FSK, the real portion of this array represents the frequency deviation of the symbol used to synchronize the bit stream.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.
 - **EVALUATE:** sync indent returns the specified sync indent.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate PSK Synchronization Parameters (bit array)

This polymorphic instance generates synchronization parameters for PSK demodulation using a synchronization sequence directly represented by the **sync bits** array.



- PSK system parameters specifies parameter values defining the PSK system. Wire the PSK system parameters cluster of the PSK (M) or PSK (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.
 - **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.
 - **differential PSK** specifies the status of the differential PSK.

disable (0)	Does not allow bit sequence encoding.
enable (1)	Allows bit sequence encoding.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase- shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from

the ideal symbol timing for I. **offset** is currently only supported for M = 4.

- **Sync bits** specifies an array of synchronization bits used to generate the **PSK synchronization parameter** cluster. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can wire a 16-bit array with the above entries to the **sync bits** parameter. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.
 - Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.
- **expected sync location** specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1. The default is 10.
- **Sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more

information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

PSK synchronization parameters returns synchronization parameters to pass to the demodulation VI.

- Note The VI always uses normal PSK when creating the **PSK synchronization parameters** cluster, even if the **PSK type** specified when creating the **PSK system parameters** cluster is **shifted** or **offset**. This supplies an absolute phase reference for the <u>MT Demodulate PSK</u> VI during the process of symbol synchronization.
- expected sync location returns the specified expected sync location.
- **sync sequence** returns the mapped PSK symbol array from the specified synchronization bits or synchronization numbers.
- **sync location uncertainty** returns the specified **sync location uncertainty**.
- **Implied sync indent** returns the specified **sync indent**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel

error out indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate PSK Synchronization Parameters (number array)

This polymorphic instance generates synchronization parameters for PSK demodulation using a synchronization sequence defined by a **sync numbers** array in which each number represents a group of bits.



PSK system parameters specifies parameter values defining the PSK system. Wire the PSK system parameters cluster of the PSK (M) or PSK (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.
- differential PSK specifies the status of the differential PSK.

disable (0) Does not allow bit sequence encoding. **enable** (1) Allows bit sequence encoding.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase- shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift
keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from the ideal symbol timing for I. **offset** is currently only supported for M = 4.

- **sync numbers** specifies an array of synchronization numbers in which the binary representation of each number is used to generate the sync bit array. By default, the internal conversion of each sync number into the corresponding synchronization bit sequence is performed with the most significant bit first (left-aligned) approach. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can specify **sync numbers** as an array with a single element xFFFE. Specify the **bits per sync number** value in this case to be 16. You can also specify **sync numbers** as an array of two elements (xFF and xFE), and specify the **bits per sync number** value as 8. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.
 - Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.
- **bits per sync number** specifies the number of bits corresponding to each sync number in the **sync numbers** array. The default is 32.
- **expected sync location** specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1. The default is 10.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **PSK synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - Note The VI always uses normal PSK when creating the **PSK synchronization parameters** cluster, even if the **PSK type** specified when creating the **PSK system parameters** cluster is **shifted** or **offset**. This supplies an absolute phase reference for the <u>MT Demodulate PSK</u> VI during the process of symbol synchronization.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped PSK symbol array from the specified synchronization bits or synchronization numbers.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.
 - **sync indent** returns the specified **sync indent**.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate QAM Synchronization Parameters (bit array)

This polymorphic instance generates synchronization parameters for QAM demodulation using a synchronization sequence directly represented by the **sync bits** array.

QAM system parameters sync bits error in (no error) expected sync location (sym sync location uncertainty (sync ident (symbols)	QAM synchronization parameters
sync indent (symbols) —	

QAM system parameters specifies parameter values defining the QAM system. Wire the QAM system parameters cluster returned by the QAM (M) or QAM (Map) instance of the <u>MT Generate</u> <u>System Parameters</u> VI to this cluster. Do not alter the values.

- samples per symbol specifies the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the samples per symbol value must be an even number. The default is 16.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^N, where N is the number of bits per symbol. The length of the vector for the symbol(s) farthest from the origin must be 1.

sync bits specifies an array of synchronization bits used to generate the QAM synchronization parameter cluster. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can wire a 16-bit array with the above entries to the sync bits parameter. For reliable symbol synchronization, specify an array of length (10 × bits per symbol). The default is 0.



Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.

expected sync location specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The

default is -1.

- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1. The default is 10.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **QAM synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped QAM symbol array from the specified synchronization bits or synchronization numbers.

- sync location uncertainty returns the specified sync location uncertainty.
- **Implied sync indent** returns the specified **sync indent**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate QAM Synchronization Parameters (number array)

This polymorphic instance generates synchronization parameters for QAM demodulation using a synchronization sequence defined by a **sync numbers** array in which each number represents a group of bits.



QAM system parameters specifies parameter values defining the QAM system. Wire the QAM system parameters cluster returned by the QAM (M) or QAM (Map) instance of the <u>MT Generate</u> <u>System Parameters</u> VI to this cluster. Do not alter the values.

- **samples per symbol** specifies the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** value must be an even number. The default is 16.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^N, where N is the number of bits per symbol. The length of the vector for the symbol(s) farthest from the origin must be 1.

sync numbers specifies an array of synchronization numbers in which the binary representation of each number is used to generate the sync bit array. By default, the internal conversion of each sync number into the corresponding synchronization bit sequence is performed with the most significant bit first (left-aligned) approach. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can specify **sync numbers** as an array with a single element xFFFE. Specify the **bits per sync number** value in this case to be 16. You can also specify **sync numbers** as an array of two elements (xFF and xFE), and specify the **bits per sync number** value as 8. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.

Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.

- **bits per sync number** specifies the number of bits corresponding to each sync number in the **sync numbers** array. The default is 32.
- expected sync location specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.

- **QAM synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped QAM symbol array from the specified synchronization bits or synchronization numbers.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.
 - **Implied sync indent** returns the specified **sync indent**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate MSK Synchronization Parameters (bit array)

This polymorphic instance generates synchronization parameters for MSK demodulation using a synchronization sequence directly represented by the **sync bits** input array.



- **Sync bits** specifies an array of synchronization bits used to generate the **MSK synchronization parameter** cluster. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can wire a 16-bit array with the above entries to the **sync bits** parameter. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.
 - Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.
- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.
- **MSK system parameters** specifies parameter values defining the MSK system. Wire the **MSK system parameters** cluster returned by the MSK instance of the <u>MT Generate System Parameters</u> VI to this cluster. Do not alter the values.
 - differential encoding enable specifies whether the bit sequence is differentially encoded.

disable (0)	Does not allow bit sequence encoding.	
enable (1)	Allows bit sequence encoding. This is the default	
	value.	

samples per symbol specifies the number of samples per symbol in the modulated output complex waveform. The default is 16.

For error-free operation, the samples per symbol should be

an even number. Applying <u>Carson's rule</u> for MSK modulation [*peak frequency deviation* = $0.25 \times symbol rate$ (Hz)], the 98% MSK bandwidth is given by $0.5 \times (1 + r) \times symbol rate + peak$ *frequency deviation*, where $0 < r \le 1$ is the filter parameter. To satisfy Nyquist criterion, a good guideline is given by *samples per symbol* = $2 \times ceil(1.5 \times bandwidth/symbol rate)$ to obtain $3 \times oversampling$.

- **expected sync location** specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- sync location uncertainty specifies the number of symbols before or after the expected sync location where the first symbol of the sync sequence may be located. This parameter is ignored if the expected sync location parameter is set to -1.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- MSK synchronization parameters returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped symbol pattern used to synchronize the bit stream. For MSK, the real portion of the mapped symbols represents the frequency deviation of the symbol value, and the imaginary portion is 0.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.

Implied sync indent returns the specified **sync indent**.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate MSK Synchronization Parameters (number array)

This polymorphic instance generates synchronization parameters for MSK demodulation using a synchronization sequence defined by a **sync numbers** input array in which each number represents a group of bits.



- **sync numbers** specifies an array of synchronization numbers in which the binary representation of each number is used to generate the sync bit array. By default, the internal conversion of each sync number into the corresponding synchronization bit sequence is performed with the most significant bit first (left-aligned) approach. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can specify **sync numbers** as an array with a single element xFFFE. Specify the **bits per sync number** value in this case to be 16. You can also specify **sync numbers** as an array of two elements (xFF and xFE), and specify the **bits per sync number** value as 8. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.
 - Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.
- **bits per sync number** specifies the number of bits corresponding to each sync number in the **sync numbers** array. The default is 32.
- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.
- **MSK system parameters** specifies parameter values defining the MSK system. Wire the **MSK system parameters** cluster returned by the MSK instance of the <u>MT Generate System Parameters</u> VI to this cluster. Do not alter the values.

differential encoding enable specifies whether the bit sequence is differentially encoded.

disable (0)	Does not allow bit sequence encoding.	
enable (1)	Allows bit sequence encoding. This is the default	
	value.	

samples per symbol specifies the number of samples per symbol in the modulated output complex waveform. The default is 16.

For error-free operation, the **samples per symbol** should be an even number. Applying <u>Carson's rule</u> for MSK modulation [*peak frequency deviation* = $0.25 \times$ *symbol rate* (Hz)], the 98% MSK bandwidth is given by $0.5 \times (1 + r) \times$ *symbol rate* + *peak frequency deviation*, where $0 < r \le 1$ is the filter parameter. To satisfy Nyquist criterion, a good guideline is given by *samples per symbol* = $2 \times$ ceil($1.5 \times$ *bandwidth/symbol rate*) to obtain $3 \times$ oversampling.

- **expected sync location** specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **MSK synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped symbol pattern used to synchronize the bit stream. For MSK, the real portion of the mapped symbols represents the frequency deviation of the symbol value, and the imaginary portion is 0.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.
 - **sync indent** returns the specified **sync indent**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel

error out indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate PAM Synchronization Parameters (bit array)

This polymorphic instance generates synchronization parameters for PAM demodulation using a synchronization sequence directly represented by the **sync bits** input array.

PAM system parameters	0101101 PAM synchronization param	eters
sync bits 🚽	Course .	
error in (no error) 🚥	error out	
expected sync location (sym		
sync location uncertainty (
sync indent (symbols)		

- PAM system parameters specifies parameter values defining the PAM system. Wire the PAM system parameters cluster returned by the PAM (M) or PAM (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.
 - samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of PAM levels in the array is 2^N, where N is the number of bits per symbol. The vector length for the symbol(s) farthest from the origin is 1.
- **Sync bits** specifies an array of synchronization bits used to generate the **PAM synchronization parameter** cluster. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can wire a 16-bit array with the above entries to the **sync bits** parameter. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.

M

Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.

expected sync location specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.

sync location uncertainty specifies the number of symbols before

or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.

sync indent specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **PAM synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped PAM symbol array from the specified synchronization bits or synchronization numbers.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.

sync indent returns the specified **sync indent**.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate PAM Synchronization Parameters (number array)

This polymorphic instance generates synchronization parameters for PAM demodulation using a synchronization sequence defined by a **sync numbers** input array in which each number represents a group of bits.



PAM system parameters specifies parameter values defining the PAM system. Wire the **PAM system parameters** cluster returned by the PAM (M) or PAM (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of PAM levels in the array is 2^N, where N is the number of bits per symbol. The vector length for the symbol(s) farthest from the origin is 1.

sync numbers specifies an array of synchronization numbers in which the binary representation of each number is used to generate the sync bit array. By default, the internal conversion of each sync number into the corresponding synchronization bit sequence is performed with the most significant bit first (left-aligned) approach. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can specify **sync numbers** as an array with a single element xFFFE. Specify the **bits per sync number** value in this case to be 16. You can also specify **sync numbers** as an array of two elements (xFF and xFE), and specify the **bits per sync number** value as 8. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.

Note To prevent false synchronization, select a bit pattern

with low correlation to nonsynchronized portions of the data stream.

- **bits per sync number** specifies the number of bits corresponding to each sync number in the **sync numbers** array. The default is 32.
- expected sync location specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - source describes the origin of the error or warning. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

- **PAM synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped PAM symbol array from the specified synchronization bits or synchronization numbers.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.
 - **Implied sync indent** returns the specified **sync indent**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate ASK Synchronization Parameters (bit array)

This polymorphic instance generates synchronization parameters for ASK demodulation using a synchronization sequence directly represented by the **sync bits** array.



- ASK system parameters specifies parameter values defining the ASK system. Wire the ASK system parameters cluster returned by the ASK instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.
 - **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of ASK levels in the array is 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin is 1.
- sync bits specifies an array of synchronization bits used to generate the ASK synchronization parameter cluster. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can wire a 16-bit array with the above entries to the sync bits parameter. For reliable symbol synchronization, specify an array of length (10 × bits per symbol). The default is 0.
 - Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.
- **expected sync location** specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.

- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **ASK synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped ASK symbol array from the specified synchronization bits or synchronization numbers.
 - **E332** sync location uncertainty returns the specified sync

location uncertainty.

EVALUATE: sync indent returns the specified sync indent.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate ASK Synchronization Parameters (number array)

This polymorphic instance generates synchronization parameters for ASK demodulation using a synchronization sequence defined by a **sync numbers** array in which each number represents a group of bits.



ASK system parameters specifies parameter values defining the ASK system. Wire the ASK system parameters cluster returned by the ASK instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of ASK levels in the array is 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin is 1.
- **sync numbers** specifies an array of synchronization numbers in which the binary representation of each number is used to generate the sync bit array. By default, the internal conversion of each sync number into the corresponding synchronization bit sequence is performed with the most significant bit first (left-aligned) approach. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can specify **sync numbers** as an array with a single element xFFFE. Specify the **bits per sync number** value in this case to be 16. You can also specify **sync numbers** as an array of two elements (xFF and xFE), and specify the **bits per sync number** value as 8. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.

Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.

bits per sync number specifies the number of bits corresponding to each sync number in the **sync numbers** array. The default is 32.

- **expected sync location** specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.

- **ASK synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the specified expected sync location.
 - **sync sequence** returns the mapped ASK symbol array from the specified synchronization bits or synchronization numbers.
 - **sync location uncertainty** returns the specified **sync location uncertainty**.
 - **Implies Sync indent** returns the specified **sync indent**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate CPM Synchronization Parameters (bit array)

This polymorphic instance generates synchronization parameters for CPM demodulation using a synchronization sequence directly represented by the **sync bits** array.

CPM system parameters	01101 CPM synchronization parameters
sync bits 🚽 🔤	
error in (no error)	
sync location uncertainty (
sync indent (symbols)	

- CPM system parameters specifies parameter values defining the CPM system. Wire the CPM system parameters cluster from the CPM (M) or CPM (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.
 - **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies the number of samples per symbol in the modulated output complex waveform. It should be an even number.
- **Sync bits** specifies an array of synchronization bits used to generate the **CPM synchronization parameter** cluster. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can wire a 16-bit array with the above entries to the **sync bits** parameter. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.



Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data stream.

- expected sync location specifies the expected location of the synchronization parameters in symbols. Set this value to -1 to search the entire waveform for the synchronization symbols. The default is -1.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the

sync sequence may be located. This parameter is ignored if the **expected sync location** parameter is set to -1. The default is 10.

- **sync indent** specifies the distance that the **sync sequence** is indented into the information block. The distance is the number of demodulated symbols preceding the **sync sequence**. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols. The default is 0.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- ESSI CPM synchronization parameters returns synchronization parameters to pass to the demodulation VI.
 - expected sync location returns the expected location of the first symbol of the synchronization sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync sequence** returns the mapped symbol pattern. Although the data type is complex, only the real portion is used. For

CPM, the real portion of the mapped symbols is the <u>frequency</u> <u>deviation</u> of the symbol value, and the imaginary portion is 0. To prevent false synchronization, configure this pattern so that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty, the signal is still demodulated.

- **sync location uncertainty** returns the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **Sync indent** returns the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate CPM Synchronization Parameters (number array)

This polymorphic instance generates synchronization parameters for CPM demodulation using a synchronization sequence defined by a **sync numbers** array in which each number represents a group of bits.



CPM system parameters specifies parameter values defining the CPM system. Wire the CPM system parameters cluster from the CPM (M) or CPM (Map) instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.

- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies the number of samples per symbol in the modulated output complex waveform. It should be an even number.

sync numbers specifies an array of synchronization numbers in which the binary representation of each number is used to generate the sync bit array. By default, the internal conversion of each sync number into the corresponding synchronization bit sequence is performed with the most significant bit first (left-aligned) approach. For example, if you specify a preamble word of 1111 1111 1111 1110 for symbol synchronization, you can specify **sync numbers** as an array with a single element xFFFE. Specify the **bits per sync number** value in this case to be 16. You can also specify **sync numbers** as an array of two elements (xFF and xFE), and specify the **bits per sync number** value as 8. For reliable symbol synchronization, specify an array of length (10 × *bits per symbol*). The default is 0.



Note To prevent false synchronization, select a bit pattern with low correlation to nonsynchronized portions of the data

stream.

- **bits per sync number** specifies the number of bits corresponding to each sync number in the **sync numbers** array. The default is 32.
- expected sync location specifies the expected location of the first symbol of the synchronization sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **sync indent** specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more

information about the error.

- **CPM synchronization parameters** returns synchronization parameters to pass to the demodulation VI.
 - **EXAMPLE** expected sync location returns the expected location of the first symbol of the synchronization sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync sequence** returns the mapped symbol pattern. Although the data type is complex, only the real portion is used. For CPM, the real portion of the mapped symbols is the <u>frequency</u> <u>deviation</u> of the symbol value, and the imaginary portion is 0. To prevent false synchronization, configure this pattern so that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this parameter is left empty, the signal is still demodulated.
 - **Sync location uncertainty** returns the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
 - **Sync indent** returns the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for

more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate System Parameters VI

Calculates parameters for use with modulation and demodulation VIs. The **system parameters** cluster from this VI wires to the corresponding parameter of the appropriate demodulation or modulation VI.

Use the pull-down menu to select an instance of this VI.

Select an instance

•
MT Generate FSK System Parameters (M)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate FSK</u> VI or the <u>MT Demodulate FSK</u> VI. This instance accepts an *M*-ary value that specifies a predefined symbol map with the number of distinct symbol map values to use as symbols.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **M-FSK** specifies the *M*-ary number, which is the number of distinct frequency deviations to use as symbols. This value must be a positive power of 2. The default is 2.
- **FSK deviation** specifies the maximum FSK frequency deviation. At <u>baseband</u> frequencies, deviations for individual symbols are evenly spaced in the interval $[-f_d, f_d]$, where f_d represents the frequency deviation. The default is 15,000.
- **symbol phase continuity** specifies whether the <u>phase</u> transitions between symbols are continuous.

With discontinuous phase-FSK (DPFSK), <u>modulation</u> consists of selecting the appropriate sinusoid based on the input data. Thus when switching between symbols, there is a discontinuity in the FSK signal phase. To emulate a hardware-based DPFSK source, this VI maintains the phase of each independent sinusoid versus time. In this way, the DPFSK modulator acts like a hardware-based (multiple switched tone generator) FSK modulator.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, DPFSK.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **FSK system parameters** returns parameter values defining the FSK system. Wire this cluster to the corresponding **system parameters** cluster of the <u>MT Modulate FSK</u> VI or the <u>MT</u> <u>Demodulate FSK</u> VI.
 - **Samples per symbol** returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** should be an even number. Applying <u>Carson's rule</u> for FSK modulation, the 98% FSK bandwidth is given by $0.5 \times (1 + r) \times symbol$ rate + peak frequency deviation, where $0 \le r \le 1$ is the filter parameter. To satisfy Nyquist criterion, a good guideline is given by samples per symbol = $2 \times ceil(1.5 \times bandwidth/symbol$ rate) to obtain $3 \times oversampling$.
 - **symbol map** returns an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of FSK levels in the array must be 2^{*N*}, where *N* is the number of bits per symbol.
 - **symbol phase continuity** returns the symbol phase state as continuous (default) or discontinuous.

- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-FSK, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate FSK System Parameters(map)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate FSK</u> VI or the <u>MT Demodulate FSK</u> VI. This VI accepts an input array of **symbol values** that explicitly define the positions of the symbol map.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol values** specifies an array of symbol values with an order that corresponds to the symbol map. The number of FSK levels specified here must be 2^N, where N is the number of bits per symbol.

This VI expects an array of integers for the **symbol values** parameter. The integers 0 through (M-1) must all be included only once in the **symbol values** array, where M is the M-ary number of the modulation. The symbol locations (FSK frequencies) are evenly spaced between -FSK Deviation and +FSK Deviation, inclusive, with the binary representation (LSB first convention) of the integers that populate the **symbol values** array that corresponds to the placement of the M-ary bits on the I/Q constellation.

For example, for 4-FSK, if you specify **FSK deviation** as 150k and **symbol values** as [0 1 3 2], then:

- 00 (symbol value 0) corresponds to -150k
- 01 (symbol value 1) corresponds to -50k
- 11 (symbol value 3) corresponds to 50k
- 10 (symbol value 2) corresponds to 150k

Therefore the generated FSK symbol map reads [–150k –50k 150k 50k]. Similarly, if you specify the **symbol values** array as [0 1 2 3], the generated FSK symbol map reads [–150k –50k 50k 150k].

FSK deviation specifies the maximum FSK frequency deviation. At <u>baseband</u> frequencies, deviations for individual symbols are evenly

spaced in the interval $[-f_d, f_d]$, where f_d represents the frequency deviation. The default is 15,000.

symbol phase continuity specifies whether the phase transitions between symbols are continuous.

With discontinuous phase-FSK (DPFSK), <u>modulation</u> consists of selecting the appropriate sinusoid based on the input data. Thus when switching between symbols, there is a discontinuity in the FSK signal phase. To emulate a hardware-based DPFSK source, this VI maintains the phase of each independent sinusoid versus time. In this way, the DPFSK modulator acts like a hardware-based (multiple switched tone generator) FSK modulator.

continuous (0)	Specifies continuous phase transitions between symbols. This is the default value.
discontinuous (1)	Specifies discontinuous phase transitions between symbols, that is, DPFSK.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

FSK system parameters returns parameter values defining the

FSK system. Wire this cluster to the corresponding **system parameters** cluster of the <u>MT Modulate FSK</u> VI or the <u>MT</u> <u>Demodulate FSK</u> VI.

- **Samples per symbol** returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** should be an even number. Applying <u>Carson's rule</u> for FSK modulation, the 98% FSK bandwidth is given by $0.5 \times (1 + r) \times symbol$ rate + peak frequency deviation, where $0 \le r \le 1$ is the filter parameter. To satisfy Nyquist criterion, a good guideline is given by samples per symbol = $2 \times ceil(1.5 \times bandwidth/symbol$ rate) to obtain $3 \times oversampling$.
- **symbol map** returns an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of FSK levels in the array must be 2^{*N*}, where *N* is the number of bits per symbol.
- **symbol phase continuity** returns the symbol phase state as continuous (default) or discontinuous.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-FSK, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.

MT Generate PSK System Parameters(M)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate PSK</u> VI or the <u>MT Demodulate PSK</u> VI. This VI generates the symbol map for <u>PSK</u>-modulated systems. It accepts an M-ary value that specifies a predefined symbol map with the number of distinct symbol map values to use as symbols.

Example Symbol Map for PSK



This symbol map corresponds to input symbol values {0, 1, 3, 2} for a quadrature-phase-shift-keying (QPSK) modulation system.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **M-PSK** specifies the *M*-ary number, which is the number of distinct states that represent symbols in the complex <u>baseband</u> modulated waveform. This value must be a positive power of 2. The default is 4.
- **differential PSK** specifies how the PSK modulation represents symbols. Differential operation is used to implement PSK formats such as differential quadrature PSK (DQPSK) and π/4-DQPSK.

disable (0)	Symbols are represented as constellation points. This is the default value.
enable (1)	Symbols are represented as the transitions between constellation points.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase-shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from the ideal symbol timing for I. offset is currently only supported for $M = 4$.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

PSK system parameters returns parameter values defining the PSK system. Wire this cluster to the corresponding system parameters cluster of the <u>MT Modulate PSK</u> VI or the <u>MT</u> <u>Demodulate PSK</u> VI. **samples per symbol** returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** should be an even number.

- **symbol map** returns an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.
- **differential PSK** returns the status of the differential PSK as enabled or disabled.
- **PSK type** returns the type of PSK modulation.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-PSK, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate PSK System Parameters(map)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate PSK</u> VI or the <u>MT Demodulate PSK</u> VI. This VI generates the symbol map for <u>PSK</u>-modulated systems. It accepts an array of **symbol values** that explicitly specifies the positions of the symbol map.

Example Symbol Map for PSK



This symbol map corresponds to input symbol values {0, 1, 3, 2} for a quadrature-phase-shift-keying (QPSK) modulation system.





symbol values specifies an array of symbol values with an order that corresponds to the symbol map. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.

This VI expects an array of integers for the **symbol values** parameter. The integers 0 through (*M*–1) must all be included only once in the **symbol values** array, where *M* is the *M*-ary number of the modulation. The binary representation (LSB first convention) of the integers represents the bit pattern that is mapped to the corresponding symbol location in the **symbol values** array. The symbol locations are generated with equal angles between all *M*-ary symbol locations and a distance of unity from the origin.

For example, in 8-PSK, if you specify **symbol values** as [5 4 2 7 3 0 1 6], the generated PSK symbol map is [(-0.383 - 0.924i), (0.383)

-0.924i, (-0.383 + 0.924i), (-0.924 - 0.383i), (0.383 + 0.924i), (0.924 + 0.383i), (0.924 - 0.383i), (-0.924 + 0.383i)].

differential PSK specifies how the PSK modulation represents symbols. Differential operation is used to implement PSK formats such as differential quadrature PSK (DQPSK) and π/4-DQPSK.

disable (0)	Symbols are represented as constellation points. This is the default value.
enable (1)	Symbols are represented as the transitions between constellation points.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase-shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by $1/2$ of a symbol period from the ideal symbol timing for I. offset is currently only supported for $M = 4$.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about

the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- PSK system parameters returns parameter values defining the PSK system. Wire this cluster to the corresponding system parameters cluster of the <u>MT Modulate PSK</u> VI or the <u>MT</u> <u>Demodulate PSK</u> VI.
 - samples per symbol returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the samples per symbol should be an even number.
 - **symbol map** returns an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N , where *N* is the number of bits per symbol.
 - **differential PSK** returns the status of the differential PSK as enabled or disabled.
 - **PSK type** returns the type of PSK modulation.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-PSK, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about

the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate QAM System Parameters(M)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate QAM</u> VI or the <u>MT Demodulate QAM</u> VI. It accepts an *M*-ary value that specifies a predefined symbol map with the number of distinct symbol map values to use as symbols.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **M-QAM** specifies the *M*-ary number, which is the number of distinct states that represent symbols in the complex <u>baseband</u> modulated waveform. This value must be a positive power of 2 and must be less than 256. The default is 16.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **QAM system parameters** returns parameter values defining the

QAM system. Wire this cluster to the corresponding **system parameters** cluster of the <u>MT Modulate QAM</u> VI or <u>MT Demodulate</u> <u>QAM</u> VI.

- **samples per symbol** returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** should be an even number.
- **symbol map** returns an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-QAM, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate QAM System Parameters(map)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate QAM</u> VI or the <u>MT Demodulate QAM</u> VI. It accepts an input array of **symbol values** that explicitly specifies the positions of the symbol map.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol values** specifies an array of symbol values with an order that corresponds to the symbol map. The number of QAM states in the array must be 2^{*N*}, where *N* is the number of bits per symbol. The length, or magnitude, of the vector for the symbols farthest from the origin must be 1.

This VI expects a two-dimensional, square array of integers for the **symbol values** parameter. 4-QAM expects a 2×2 array (2 rows, 2 columns), 16-QAM expects a 4×4 array, and so on. The integers 0 through (*M*–1) must all be included only once in the **symbol values** array, where *M* is the *M*-ary number of the modulation. *M* must be less than 256. The binary representation (LSB first convention) of the integers represents the bit pattern that is mapped to the corresponding symbol location in the **symbol values** array.

For example, for 4-QAM, if you specify symbol values as

[3 1]

[0 2]

the generated QAM symbol map is [(-0.707 - 0.707i), (0.707 + 0.707i), (0.707 - 0.707i), (-0.707 + 0.707i)].

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- QAM system parameters returns parameter values defining the QAM system. Wire this cluster to the corresponding system parameters cluster of the <u>MT Modulate QAM</u> VI or <u>MT Demodulate</u> <u>QAM</u> VI.
 - **samples per symbol** returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** should be an even number.
 - **symbol map** returns an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^N , where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-QAM, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

Status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate MSK System Parameters

This polymorphic instance calculates parameters for use with either the <u>MT Modulate MSK</u> VI or the <u>MT Demodulate MSK</u> VI.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- differential encoding enable specifies whether to enable differential encoding of the bit stream.

disable (0) Does not allow bit stream encoding.

enable (1) Allows bit stream encoding. This is the default value.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- MSK system parameters returns parameter values defining the MSK system. Wire this parameter to the corresponding system

parameters cluster of the <u>MT Modulate MSK</u> VI or the <u>MT</u> <u>Demodulate MSK</u> VI.

- **differential encoding enable** specifies whether the bit sequence is differentially encoded.
- **Samples per symbol** returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** should be an even number. Applying <u>Carson's rule</u> for MSK modulation [*peak frequency deviation* = $0.25 \times$ *symbol rate* (Hz)], the 98% MSK bandwidth is given by $0.5 \times (1 + r) \times$ *symbol rate* + *peak frequency deviation*, where $0 < r \le 1$ is the filter parameter. To satisfy Nyquist criterion, a good guideline is given by *samples per symbol* = $2 \times$ ceil($1.5 \times$ *bandwidth/symbol rate*) to obtain $3 \times$ oversampling.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-MSK, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate PAM system parameters (M)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate PAM</u> VI or the <u>MT Demodulate PAM</u> VI. It accepts an *M*-ary value that specifies a predefined symbol map with the number of distinct levels to use as symbols.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **M-PAM** specifies the *M*-ary number, which for PAM is the number of distinct states that represent symbols in the complex baseband modulated waveform. This value must be a positive power of 2. The default is 2.
 - Note The *M*-to-symbol-map mapping is performed using a gray-coding approach. In this approach, the binary interpretation of the index of any value in the symbol map array corresponds to the bit sequence associated with the symbol map location. For example, with 4-PAM (2 bits per symbol), a returned symbol map in the order [-1, -0.33, 1.00, 0.33] implies that the bits 00 are mapped to the symbol map location –1, the bits 01 are mapped to the symbol map location 1.00, and the bits 11 are mapped to the symbol map location 0.33.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain**

Error or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- PAM system parameters returns parameter values defining the PAM system. Wire this cluster to the corresponding system parameters cluster of the <u>MT Modulate PAM</u> VI or the <u>MT</u> <u>Demodulate PAM</u> VI.
 - **samples per symbol** returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** should be an even number.
 - **symbol map** returns an ordered array that maps each symbol to its desired level. The number of PAM levels in the array is 2^N , where *N* is the number of bits per symbol. The length of the vector for the symbol or symbols farthest from the origin is 1.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-PAM, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate PAM system parameters (map)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate PAM</u> VI or the <u>MT Demodulate PAM</u> VI. It accepts an array of **symbol values** that explicitly specifies the positions of the symbol map with symbols that are evenly distributed between –1 and 1.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol values** specifies an array of symbol values with an indexbased order that corresponds to the symbol map. The length of the **symbol values** array must be a positive power of 2, and the entries must be unique and in the range 0 to *M*-1, where *M* is the length of the **symbol values** array.

This VI expects an array of integer for the **symbol values** parameter. The integers 0 through (*M*-1) must all be included only once in the symbol values array, where *M* is the *M*-ary number of the modulation. The binary representation (LSB first convention) of the integers represents the bit pattern that is mapped to the corresponding symbol location in the **symbol values** array. The symbol locations are evenly spaced between (-1 + 0i) and (1 + 0i), inclusive, along the real axis (I axis).

For example, for 4-PAM, if you specify **symbol values** as $[2 \ 3 \ 0 \ 1]$, the generated PAM symbol map is [(0.333 + 0i), (1 + 0i), (-1 + 0i), (-0.333 + 0i)].

Note An entry at index *n* is mapped to the PAM symbol map at location 2n/(M - 1) - 1, where *M* is the size of the PAM constellation. For example, if you specify a **symbol values** array with entries [0, 1, 2, 3], the symbol-to-symbol pairing is *symbol* $0 \rightarrow$ level -1, *symbol* $1 \rightarrow$ level -1/3, *symbol* $2 \rightarrow$ level 1/3, *symbol* $3 \rightarrow$ level 1.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- PAM system parameters returns parameter values defining the PAM system. Wire this cluster to the corresponding system parameters cluster of the <u>MT Modulate PAM</u> VI or the <u>MT</u> <u>Demodulate PAM</u> VI.
 - samples per symbol returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the samples per symbol should be an even number.
 - **symbol map** returns an ordered array that maps each symbol to its desired level. The number of PAM levels in the array is 2^N , where *N* is the number of bits per symbol. The length of the vector for the symbol or symbols farthest from the origin is 1.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-PAM, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and

select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate ASK system parameters (M)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate ASK</u> VI or the <u>MT Demodulate ASK</u> VI. It accepts an *M*-ary value that specifies a predefined symbol map with the number of distinct levels to use as symbols.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **M-ASK** specifies the *M*-ary number, which for <u>ASK</u> is the number of distinct states that represent symbols in the complex baseband modulated waveform. This value must be a positive power of 2.
 - Note The *M*-to-symbol-map mapping is performed using a gray-coding approach. In this approach, the binary interpretation of the index of any value in the symbol map array corresponds to the bit sequence associated with the symbol map location. For example, with 4-ASK (2 bits per symbol), a returned symbol map in the order [0, 0.33, 1.00, 0.67] implies that the bits 00 are mapped to the symbol map location 0, the bits 01 are mapped to the symbol map location 0.33, the bits 11 are mapped to the symbol map location 0.67, and the bits 10 are mapped to the symbol map location 1.00. The default is 2.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more

information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

ASK system parameters returns parameter values defining the ASK system. Wire this cluster to the corresponding system parameters cluster of the MT Modulate ASK VI or the MT Demodulate ASK VI.

- **samples per symbol** returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the **samples per symbol** should be an even number.
- **symbol map** returns an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of ASK levels in the array must be 2^N, where N is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-ASK, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel

error out indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate ASK system parameters (map)

This polymorphic instance calculates parameters for use with either the <u>MT Modulate ASK</u> VI or the <u>MT Demodulate ASK</u> VI. It accepts an array of **symbol values** that explicitly specifies the symbol positions of a symbol map with symbols that are evenly distributed between 0 and 1.



- **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol values** specifies an array of symbol values with an indexbased order that corresponds to the symbol map. The length of the **symbol values** array must be a positive power of 2, and the entries must be unique and in the range 0 to *M*-1, where *M* is the length of the **symbol values** array.

This VI expects an array of integers for the **symbol values** parameter. The integers 0 through (M-1) must all be included only once in the symbol values array, where M is the M-ary number of the modulation. The binary representation (LSB first convention) of the integers represents the bit pattern that is mapped to the corresponding symbol location in the **symbol values** array. The symbol locations are evenly spaced between (0 + 0i) and (1 + 0i), inclusive, along the real axis (I axis).

For example, for 4-ASK, if you specify **symbol values** as $[2 \ 3 \ 0 \ 1]$, the generated ASK symbol map is [(0.667 + 0i), (1 + 0i), (0 + 0i), (0.333 + 0i)].

Note An entry at index *n* is mapped to the ASK symbol map at location n/(M-1), where *M* is the size of the ASK constellation. For example, if you specify **symbol values** with entries [0, 1, 2, 3], the symbol-to-symbol pairing is *symbol* $0 \rightarrow$ level 0, *symbol* $1 \rightarrow$ level 1/3, *symbol* $2 \rightarrow$ level 2/3, *symbol* $3 \rightarrow$ level 1.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- ASK system parameters returns parameter values defining the ASK system. Wire this cluster to the corresponding system parameters cluster of the MT Modulate ASK VI or the MT Demodulate ASK VI.
 - samples per symbol returns the number of samples per symbol in the modulated output complex waveform. For errorfree operation, the samples per symbol should be an even number.
 - **symbol map** returns an ordered array that maps each Boolean symbol to its desired deviation frequency. The number of ASK levels in the array must be 2^N, where N is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 16-ASK, *M*=16).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and

select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Generate CPM System Parameters (M)

This polymorphic instance accepts an *M*-ary value that specifies a predefined symbol map with the number of distinct symbol map values to use as symbols.



- **samples per symbol** specifies the number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **M-CPM** specifies the M-ary number, which is the number of distinct frequency deviations to use as symbols. This value must be a positive power of 2.
- **modulation index, h(i)** specifies the modulation index for the CPM modulation scheme. Refer to <u>CPM Modulation</u> for more information about modulation indices. The modulation index should be of proper form, that is, the numerator should be lesser than the denominator. The **modulation index** may vary between symbol intervals.
 - **numerator** specifies the numerator of the **modulation index**.
 - denominator specifies the denominator of the modulation index.
- **symbol rate** specifies the desired symbol rate, in hertz.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- CPM system parameters returns parameter values defining the CPM system. Wire this control to the corresponding system parameters cluster of the <u>MT Modulate CPM</u>, <u>MT Demodulate CPM</u>, or <u>MT Detect CPM</u> VI.
 - **samples per symbol** specifies the number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate.
 - **symbol map** returns an ordered array that maps each Boolean symbol to its desired frequency deviation.
- **bits per symbol** returns the number of bits represented by each symbol. This value is equal to $Log_2(M)$, where *M* is the order of the modulation (for example, for 4-CPM, *M*=4).
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.
MT Generate CPM System Parameters (Map)

Calculates parameters for use with either the <u>MT Modulate CPM</u> VI or the <u>MT Demodulate CPM</u> VI. Wire the **CPM system parameters** passed from this VI to the corresponding cluster of the <u>MT Modulate CPM</u>, <u>MT Demodulate CPM</u>, and <u>MT Detect CPM</u> VIs. It accepts an input array of symbol values that explicitly define the positions of the symbol map.



- **samples per symbol** specifies the number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol indices** specifies an array of symbol values with an order that corresponds to the symbol map. The number of CPM levels specified here must be 2^N, where N is the number of bits per symbol.
- **modulation index, h(i)** specifies the modulation index for the CPM modulation scheme. Refer to <u>CPM Modulation</u> for more information about modulation indices. The modulation index should be of proper form, that is, the numerator should be lesser than the denominator. The **modulation index** may vary between symbol intervals.
 - **numerator** specifies the numerator of the **modulation index**.

denominator specifies the denominator of the **modulation index**.

- **symbol rate** specifies the desired symbol rate, in hertz.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- CPM system parameters returns parameter values defining the CPM system. Wire this control to the corresponding system parameters cluster of the <u>MT Modulate CPM</u>, <u>MT Demodulate CPM</u>, or <u>MT Detect CPM</u> VI.
 - **samples per symbol** specifies the number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate.
 - **symbol map** returns an ordered array that maps each Boolean symbol to its desired frequency deviation.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.

MT Map Bits to Symbols VI

Maps bits to complex valued symbols for PSK, QAM, PAM, ASK, and CPM modulation schemes and frequency deviations for FSK and MSK modulation schemes.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Map Bits to FSK Symbols

This polymorphic instance maps an incoming bit stream to symbols comprised of FSK frequency deviations.



- **input bit stream** specifies the incoming bit stream to be mapped to FSK symbols.
- **symbol map** specifies an ordered array that maps each symbol value to its desired deviation frequency. The number of FSK levels in the array must be 2^{*N*}, where *N* is the number of bits per symbol. To specify a prebuilt map, unbundle the **symbol map** element from the **FSK system parameters** cluster generated by the FSK (M) or FSK (Map) instance of the <u>MT Generate System Parameters</u> VI.
- reset? specifies whether stored state information is cleared on each call to this VI. When the input bit stream is not comprised of an integer number of symbols, the carryover bits are buffered. When reset? is set to TRUE (default), this buffer is cleared at each call. When reset? is set to FALSE, the carryover bits are added to the beginning of the input bit stream at the next call to this VI. This option is useful when the current block of data is contiguous with the preceding block of data. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain**

Warning from the shortcut menu for more information about the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbols** returns an array of FSK frequency deviations with a oneto-one mapping to the **input bit stream** based on the specified FSK **symbol map**.
 - Note Wire this parameter to the FSK instance of the <u>MT</u> <u>Pulse Shaping Filter</u> VI to generate the oversampled baseband complex waveform.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map Bits to PSK Symbols

This polymorphic instance maps an incoming bit stream to complexvalued PSK symbols.



- **input bit stream** specifies the incoming bit stream to be mapped to PSK symbols.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^N, where *N* is the number of bits per symbol. To specify a prebuilt map, unbundle the **symbol map** element from the **PSK system parameters** cluster generated by the PSK (M) or PSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI.

PSK type specifies the type of phase-shift keying modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase-shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by $1/2$ of a symbol period from the ideal symbol timing for I. offset is currently only supported for $M = 4$.

differential PSK specifies how the modulation represents symbols. Differential operation is used to implement PSK formats such as differential quadrature PSK (DQPSK) and π/4 DQPSK.

disable (0)	Symbols are represented as constellation points. This is the default value.

enable (1) Symbols are represented as the transitions between constellation points.

- reset? specifies whether stored state information is cleared on each call to this VI. When the input bit stream is not comprised of an integer number of symbols, the carryover bits are buffered. When reset? is set to TRUE (default), this buffer is cleared at each call. When reset? is set to FALSE, the carryover bits are added to the beginning of the input bit stream at the next call to this VI. This option is useful when the current block of data is contiguous with the preceding block of data. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbols** returns an array of complex-valued PSK symbols with a one-to-one mapping to the **input bit stream** based on the specified PSK **symbol map**.
 - Note When the **PSK type** is set to **shifted**, the mapped symbols are additionally multiplied by the complex phasor, as shown by the following equation:

 $e^{jm\frac{\pi}{M}}, m = 0..M - 1, \ j = \sqrt{-1}$

where *M* specifies the *M*-ary PSK format. This multiplication ensures that the generated PSK oversampled waveform never transitions through the origin after pulse shaping.



Note Wire this parameter to the <u>MT Pulse Shaping Filter</u> VI to generate the oversampled baseband complex waveform.

error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map Bits to QAM Symbols

This polymorphic instance maps an incoming bit stream to complexvalued QAM symbols.



- [18] input bit stream specifies the incoming bit stream to be mapped to QAM symbols.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1. To specify a prebuilt map, unbundle the **symbol map** element from the **QAM system parameters** cluster generated by the QAM (M) or QAM (Map) instance of the MT <u>Generate System Parameters</u> VI.
- reset? specifies whether stored state information is cleared on each call to this VI. When the input bit stream is not comprised of an integer number of symbols, the carryover bits are buffered. When reset? is set to TRUE (default), this buffer is cleared at each call. When reset? is set to FALSE, the carryover bits are added to the beginning of the input bit stream at the next call to this VI. This option is useful when the current block of data is contiguous with the preceding block of data. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbols** returns an array of complex-valued QAM symbols with a one-to-one mapping to the **input bit stream** based on the specified QAM **symbol map**.



Note Wire this parameter to the <u>MT Pulse Shaping Filter</u> VI to generate the oversampled baseband complex waveform.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map Bits to MSK Symbols

This polymorphic instance maps an incoming bit stream to symbols comprised of MSK frequency deviations.



- **input bit stream** specifies the incoming bit stream to be mapped to MSK symbols.
- **symbol rate** specifies the symbol rate, in hertz, of the **input bit stream**.
 - **Note** For MSK modulation, the frequency deviation is $\frac{1}{4}$ the symbol rate. For nondifferential MSK, the symbol-to-bit mapping is performed by mapping bit $0 \rightarrow -0.25 \times symbol$ rate, and mapping bit $1 \rightarrow +0.25 \times symbol$ rate.
- differential encoding enable specifies whether to enable differential encoding of the bit stream prior to mapping to MSK symbols.

disable (0)Does not allow bit stream encoding.enable (1)Allows bit stream encoding. This is the default value.

- reset? specifies whether the carryover bit at the end of differential encoding is added to the beginning of the input bit stream at the next call to this VI. When reset? is set to TRUE (default), the carryover bit is cleared at each call. When reset? is set to FALSE and differential coding enable is set to TRUE (default), the carryover bit is added to the beginning of the input bit stream at the next call to this VI. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE

(checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbols** returns an array of MSK frequency deviations with a one-to-one mapping to the **input bit stream**.
 - $\overline{\mathbb{N}}$

Note Wire this parameter to the MSK instance of the <u>MT</u> <u>Pulse Shaping Filter</u> VI to generate the oversampled baseband complex waveform.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map Bits to PAM Symbols

This polymorphic instance maps an incoming bit stream to complexvalued PAM symbols.



- **input bit stream** specifies the incoming bit stream to be mapped to PAM symbols.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of PAM states in the array must be 2^N, where N is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1. To specify a prebuilt map, unbundle the **symbol map** element from the **PAM system parameters** cluster generated by the PAM (M) or PAM (Map) instance of the MT <u>Generate System Parameters</u> VI.
- reset? specifies whether stored state information is cleared on each call to this VI. When the input bit stream is not comprised of an integer number of symbols, the carryover bits are buffered. When reset? is set to TRUE (default), this buffer is cleared at each call. When reset? is set to FALSE, the carryover bits are added to the beginning of the input bit stream at the next call to this VI. This option is useful when the current block of data is contiguous with the preceding block of data. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbols** returns an array of complex-valued PAM symbols with a one-to-one mapping to the **input bit stream** based on the specified PAM **symbol map**.



Note Wire this parameter to the <u>MT Pulse Shaping Filter</u> VI to generate the oversampled baseband complex waveform.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map Bits to ASK Symbols

This polymorphic instance maps an incoming bit stream to complexvalued ASK symbols.



- **input bit stream** specifies the incoming bit stream to be mapped to ASK symbols.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of ASK states in the array must be 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1. To specify a prebuilt map, unbundle the **symbol map** element from the **ASK system parameters** cluster generated by the ASK (M) or ASK (Map) instance of the MT <u>Generate System Parameters</u> VI.
- reset? specifies whether stored state information is cleared on each call to this VI. When the input bit stream is not comprised of an integer number of symbols, the carryover bits are buffered. When reset? is set to TRUE (default), this buffer is cleared at each call. When reset? is set to FALSE, the carryover bits are added to the beginning of the input bit stream at the next call to this VI. This option is useful when the current block of data is contiguous with the preceding block of data. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbols** returns an array of complex-valued ASK symbols with a one-to-one mapping to the **input bit stream** based on the specified ASK **symbol map**.



Note Wire this parameter to the <u>MT Pulse Shaping Filter</u> VI to generate the oversampled baseband complex waveform.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map Bits to CPM Symbols

This polymorphic instance maps an incoming bit stream to symbols comprised of CPM frequency deviations.



- **input bit stream** specifies the incoming bit stream to be mapped to CPM symbols.
- **symbol map** specifies an ordered array that maps each symbol value to its desired deviation frequency. To specify a prebuilt map, unbundle the **symbol map** element from the **CPM system parameters** cluster generated by the CPM instance of the <u>MT</u> <u>Generate System Parameters</u> VI.
- reset? (T) specifies whether stored state information is cleared on each call to this VI. When the input bit stream is not comprised of an integer number of symbols, the carryover bits are buffered. When reset? is set to TRUE (default), this buffer is cleared at each call. When reset? is set to FALSE, the carryover bits are added to the beginning of the input bit stream at the next call to this VI. This option is useful when the current block of data is contiguous with the preceding block of data.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about

the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **symbols** returns an array of CPM frequency deviations with a oneto-one mapping to the **input bit stream** based on the specified CPM **symbol map**.
 - N R

Note Wire this parameter to the CPM instance of the <u>MT</u> <u>Pulse Shaping Filter</u> VI to generate the oversampled baseband complex waveform.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Use this VI in conjunction with the <u>MT Generate Filter Coefficients</u> VI and the polymorphic <u>MT Pulse Shaping Filter</u> VI for generating the modulated oversampled complex baseband waveform.



Note The mapping from bits to symbols depends on the number of bits per symbol, which equals $\log_2(M)$, with *M* representing the size of the modulation scheme. For example, in a 256-QAM system, an input of 4,000 message bits is mapped to:

 $\frac{4000}{\log_2(256)} = 500 \text{ QAM symbols}$



Tip Ensure that the size of the **input bit stream** is an integer multiple of the bits per symbol to guarantee an integer number of symbols. When this is not the case, remaining bits are buffered and returned on the next call when **reset?** is set to FALSE.

MT Map Symbols to Bits VI

Maps complex valued PSK-, QAM-, PAM-, ASK-, FSK-, MSK-, and CPMmodulated symbols to an **output bit stream** based on a user-specified symbol map.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Map FSK Symbols to Bits

This polymorphic instance maps symbols comprised of FSK frequency deviations to an **output bit stream** based on a user-specified symbol map.



- **symbols** specifies an array of FSK frequency deviations for one-toone mapping to the **output bit stream** based on the **symbol map**.
- **symbol map** specifies an ordered array that maps each symbol value to its desired deviation frequency. The number of FSK levels in the array must be 2^{*N*}, where *N* is the number of bits per symbol. To specify a prebuilt map, unbundle the **symbol map** element from the **FSK system parameters** cluster generated by the FSK (M) or FSK (Map) instance of the <u>MT Generate System Parameters</u> VI.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns a bit stream with a one-to-one mapping

to the input symbols based on the specified FSK symbol map.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map PSK Symbols to Bits

This polymorphic instance maps complex-valued PSK symbols to an **output bit stream** based on a user-specified symbol map.



symbols specifies the incoming complex-valued PSK symbols to be mapped to the **output bit stream** based on the **symbol map**.

symbol map specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of PSK states in the array must be 2^{*N*}, where *N* is the number of bits per symbol. To specify a prebuilt map, unbundle the **symbol map** element from the **PSK system parameters** cluster generated by the PSK (M) or PSK (Map) instance of the <u>MT Generate System</u> Parameters VI.

PSK type specifies the type of phase-shift keying modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π /M each symbol.
offset	Sets the modulation type to offset quadrature phase-shift keying (OQPSK). This modulation scheme is a form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from the ideal symbol timing for I. offset is currently only supported for $M = 4$.

carrier phase correction specifies the carrier phase correction applied to the **symbol** array prior to mapping. The correction overcomes the starting phase ambiguity during shifted-PSK demodulation. Use this control when the PSK **modulation type** is set to **shifted** and **differential PSK** is set to **disable**. The default is 0.0. **Differential PSK** specifies how the modulation represents symbols. Differential operation is used to implement PSK formats such as differential quadrature PSK (DQPSK) and π/4 DQPSK.

disable (0)	Symbols are represented as constellation points. This is the default value.
enable (1)	Symbols are represented as the transitions between constellation points.

- reset? specifies whether stored state information is cleared on each call to this VI. When the input bit stream is not comprised of an integer number of symbols, the carryover bits are buffered. When reset? is set to TRUE (default), this buffer is cleared at each call. When reset? is set to FALSE, the carryover bits are added to the beginning of the input bit stream at the next call to this VI. This option is useful when the current block of data is contiguous with the preceding block of data. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

[18] output bit stream returns a bit stream with a one-to-one mapping

to the input symbols based on the specified PSK symbol map.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map QAM Symbols to Bits

This polymorphic instance maps complex-valued QAM symbols to an **output bit stream** based on a user-specified symbol map.



- **symbols** specifies the incoming complex-valued QAM symbols to be mapped to the **output bit stream** based on the **symbol map**.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1. To specify a prebuilt map, unbundle the **symbol map** element from the **QAM system parameters** cluster generated by the QAM (M) or QAM (Map) instance of the MT <u>Generate System Parameters</u> VI.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **(III)** output bit stream returns a bit stream with a one-to-one mapping to the input symbols based on the specified QAM symbol map.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map MSK Symbols to Bits

This polymorphic instance maps symbols comprised of MSK frequency deviations to an **output bit stream** based on a user-specified symbol rate.



- **symbols** specifies an array of MSK frequency deviations for oneto-one mapping to the **output bit stream**.
- **symbol rate** specifies the symbol rate, in hertz, of the **symbols** data.
 - **Note** For MSK modulation, the frequency deviation is $\frac{1}{4}$ the symbol rate. For nondifferential MSK, the symbol-to-bit mapping is performed by mapping bit $0 \rightarrow -0.25 \times symbol$ rate, and mapping bit $1 \rightarrow +0.25 \times symbol$ rate.
- differential encoding enable specifies whether to enable differential encoding of the input symbols prior to mapping to the output bit stream.

disable (0)	Does not allow differential encoding for input symbols.
enable (1)	Allows differential encoding for input symbols. This is the default value.

- **TEN** reset? specifies whether the carryover bit at the end of differential encoding is added to the beginning of the input bit stream at the next call to this VI. When reset? is set to TRUE (default), the carryover bit is cleared at each call. When reset? is set to FALSE and differential coding enable is set to TRUE (default), the carryover bit is added to the beginning of the input bit stream at the next call to this VI. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more

information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output bit stream returns a bit stream with a one-to-one mapping to the input symbols based on the specified MSK symbol map.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map PAM Symbols to Bits

This polymorphic instance maps complex-valued PAM symbols to an **output bit stream** based on a user-specified symbol map.



- **symbols** specifies the incoming complex-valued PAM symbols to be mapped to the **output bit stream** based on the **symbol map**.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of PAM states in the array must be 2^N, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1. To specify a prebuilt map, unbundle the **symbol map** element from the **PAM system parameters** cluster generated by the PAM (M) or PAM (Map) instance of the MT <u>Generate System Parameters</u> VI.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **(18)** output bit stream returns a bit stream with a one-to-one mapping to the input symbols based on the specified PAM symbol map.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map ASK Symbols to Bits

This polymorphic instance maps complex-valued ASK symbols to an **output bit stream** based on a user-specified symbol map.



- **symbols** specifies the incoming complex-valued ASK symbols to be mapped to the **output bit stream** based on the **symbol map**.
- **symbol map** specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of ASK states in the array must be 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin must be 1. To specify a prebuilt map, unbundle the **symbol map** element from the **ASK system parameters** cluster generated by the ASK (M) or ASK (Map) instance of the MT <u>Generate System Parameters</u> VI.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **(III)** output bit stream returns a bit stream with a one-to-one mapping to the input symbols based on the specified ASK symbol map.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Map CPM Symbols to Bits

This polymorphic instance maps symbols comprised of frequency deviations to an output bit stream based on a user-specified symbol map.



symbols specifies an array of frequency deviations for one-to-one mapping to the **output bit stream** based on the **symbol map**.

- **symbol map** specifies an ordered array that maps each symbol value to its desired deviation frequency. To specify a prebuilt map, unbundle the **symbol map** element from the **CPM system parameters** cluster generated by the CPM instance of the <u>MT</u> <u>Generate System Parameters</u> VI.
- synchronization parameters specifies parameter values describing the synchronization sequence and the range of bits over which to search for the sequence. Wire the CPM synchronization parameters cluster returned by the CPM bit array or number array instances of the MT Generate Synchronization Parameters VI to this cluster.
 - Note If the CPM synchronization parameters cluster is not wired, the demodulator does not attempt to synchronize, and the constellation of the demodulated waveform has a carrier phase ambiguity.
 - expected sync location specifies the expected location of the first symbol of the synchronization sequence. This value is an index to the input complex waveform. A value of -1 searches the entire input complex waveform and ignores the sync location uncertainty parameter.
 - **sync sequence** specifies the mapped symbol pattern. Although the data type is complex, only the real portion is used. For CPM, the real portion of the mapped symbols is the <u>frequency deviation</u> of the symbol value, and the imaginary portion is 0. To prevent false synchronization, configure this pattern so that there is a low probability of accidental correlation to nonsynchronized parts of the data stream. If this

parameter is left empty, the signal is still demodulated.

- **sync location uncertainty** specifies the number of symbols before or after the **expected sync location** where the first symbol of the **sync sequence** may be located. This parameter is ignored if the **expected sync location** parameter is set to -1.
- **sync indent** specifies the distance that the sync sequence is indented into the information block. The distance is the number of demodulated symbols preceding the sync sequence. For example, a value of 10 indicates that the **output bit stream** consists of 10 data symbols, followed by the **sync sequence**, followed by the remaining data symbols.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(III)** output bit stream returns a bit stream with a one-to-one mapping to the input symbols based on the specified CPM symbol map.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu
for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Use this VI in conjunction with the <u>MT Matched Filter</u> VI, the <u>MT Align To</u> <u>Ideal Symbols</u> VI, the <u>MT Decimate Oversampled Waveform</u> VI, and the <u>MT Calculate BER</u> VI to perform waveform analysis on the modulated waveform.



Note The mapping of symbols to bits depends on the number of bits per symbol which equals $\log_2(M)$ with *M* representing the size of the modulation scheme. For example, in a 256-QAM system, 500 QAM symbols are mapped to:

 $(500) \times \log_2(256) = 4000$ message bits

MT Open Session VI

Creates an empty modulation session refnum. When the modulation session refnum is created, navigate to the **All Functions**»**Application Control** palette and drop an empty property node on the block diagram. When you wire the modulation session refnum into the property node, the property node displays the name **Modulation Toolkit** as its title. You can either write (set) or read (get) attributes using the property node.



- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- modulation session returns the modulation session refnum that can be used to configure the behavior and operation of the appropriate Modulation Toolkit VI that accepts the modulation session refnum at its input. Use the <u>MT Close Session</u> VI to close the modulation session.
 - Note You may navigate to the Application Control subpalette, located on the Functions»Programming palette,

and place a property node on the block diagram. After you wire the modulation session refnum to the property node input, all the modulation-specific attributes are accessible.

Remember to free the modulation session refnum using the MT Close Modulation Session VI before the completion of execution to avoid possible memory overhead issues.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Pack Bits VI

Packs data to an unsigned or signed integer array. You can specify the order in which the bits are packed into the integer array either as **MSB First** or **LSB First**. You can also specify the integer format to be **Signed** or **Unsigned** using the **integer format** parameter.

This VI packs *MN*-sized binary data of the form $b_i = \{0,1\}, i=0...MN-1$ to an unsigned or signed integer array $a_n n=0...N-1$ of size *N*, where *M* denotes the number of bits per integer.

Details



- input bit stream specifies the binary bit stream b i={0,1}, i=0...MN-1 to be packed into integers. The default is empty.
- **bits per integer** specifies the number of binary data values that are packed into an integer. If you choose **Unsigned** for the **integer format**, the maximum value is 31. If you choose **Signed**, the maximum value is 32. The default is 1.
- **packed bit order** specifies the order in which the binary data stream has been packed into integers.

MSB	Data is packed with the most significant bit (MSB) first.
first (0)	This is the default value.
LSB first (1)	Data is packed with the least significant bit (LSB) first.

integer format specifies the input integer format.

Unsigned (0)) The entire number is packed as a positive integer. This is the default value.	
Signed (1)	The most significant bit (MSB) determines the sign of the input integer.	

reset ? specifies how the VI handles buffered data. When the length of the input bit stream is not a multiple of the bits per

integer value, the leftover bits are buffered inside the VI. When **reset?** is set to FALSE, these buffered bits are prepended to the **input bit stream** during the next iteration. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Output integers** returns the output integer stream $a_n n=0...N-1$ corresponding to the packed bits.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about

the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The relationship between the **output integers** array and the **input bit stream** for the Unsigned integer format is given by the following equations:

$$a_n = \sum_{j=0}^{M-1} b_{Mn+j} 2^{(M-1)-j}, n = 0...N-1$$
(packed bit order set to MSB First)
$$a_n = \sum_{j=0}^{M-1} b_{Mn+j} 2^j, n = 0...N-1$$
(packed bit order set to LSB First)

If the **integer format** is set to Signed, then the MSB at the bit position N-1 is used to determine the sign of the output integer. The sign is positive if MSB is 0 and negative if MSB is 1. For example, if you set the **integer format** to **Unsigned** and **bits per integer** as 3, and **packed bit order** as **MSB First**, the bits [1 0 0] are mapped to the integer 4. If **packed bit order** is specified as **LSB First**, the bits are mapped to the integer 1. If the **integer format** is set to **Signed** and **packed bit order** is **MSB First**, the bits [1 0 0] are packed to integer –4 because the MSB is 1 in this case. If **packed bit order** is set to **LSB First**, the bits are mapped to integer 1, because MSB is 0 in this case.

 $\overline{\mathbb{N}}$

Note This VI is continuable, meaning that you can call it for multiple iterations inside a loop with **reset?** set to FALSE. In this case, the VI buffers the leftover bits whenever the length of the **input bit stream** is not a multiple of the specified **bits per integer**. These leftover bits are prepended to the beginning of the **input bit stream** during the next iteration, provided **reset?** is set to FALSE.

MT UnPack Bits VI

Unpacks an array of signed or unsigned integer data. You can specify the **packed bit order** as **MSB First** or **LSB First**. You can also specify the **integer format** to be **Signed** or **Unsigned**.

This array unpacks data of the form a_n ,n = 0...N-1 into an output binary bit stream { b_i }, i = 0...MN - 1, where M denotes the number of bits per integer.

Details



input integers specifies the signed or unsigned integer data a_n , n=0...N-1 to be converted into binary data. The default is empty.

- **bits per integer** specifies the number of bits corresponding to the binary representation of every element in the **input integers** array. If you choose **Unsigned** for the **integer format**, the maximum value is 31. If you choose **Signed**, the maximum value is 32.The default is 1.
- **packed bit order** specifies the order in which the binary data stream has been packed into integers.

MSB	Data is packed with the most significant bit (MSB) first.
first (0)	This is the default value.
LSB first (1)	Data is packed with the least significant bit (LSB) first.

integer format specifies the input integer format.

Unsigned (0)	The entire number is packed as a positive integer. This is the default value.
Signed (1)	The most significant bit (MSB) determines the sign of the input integer.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **(iii)** output bit stream returns the output binary data stream $b_i = \{0,1\}$, i=0...MN-1 that corresponds to the packed representation of the input binary data.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The relationship between the **input integers** and the **output bit stream** is given by the following equations:

 $b_{i \times M+j} = (a_i)(M-1-j) \& 1, j = 0...M-1, i=0...N-1$ (packed bit order set to MSB First)

 $b_{i \times M+j} = (a_i \gg j) \& 1, j = 0...M - 1, i=0...N - 1$ (**packed bit order** set to LSB First)

Where » denotes the operation of right shifting, and the operator & stands for the logical AND operation. For example, assume the **input integers** array is the element [4], the **bits per integer** value is specified as 3, and **integer format** is set to **Unsigned**. If **packed bit order** is set to **MSB First**, the **output bit stream** is [1 0 0]. If the **packed bit order** is set to **LSB first**, the **output bit stream** is [0 0 1]. The signed negative numbers are stored as the 2's complement. Thus, if **integer format** is set to **Signed** and the **bits per integer** is set to **3**, the **input integer** array is the element [–3], and the **packed bit order** is set to **MSB First** the

Measurements

Owning Palette: Digital

Use the VIs on the **Measurements** palette to measure parameters including BER, Rho, burst timing, and quadrature impairments.

Palette Object	Description	
<u>MT</u> <u>Calculate</u> <u>BER</u>	Calculates the average <u>bit error rate (BER)</u> against a PN sequence or user-specified reference bit pattern.	
<u>MT</u> <u>Measure</u> Quadrature Impairments	Measures and reports quadrature impairments over a single user-specified window for each of the supported modulation types. This VI must be called once for each measurement window.	
<u>MT</u> <u>Measure</u> <u>Burst</u> Timing	Locates a burst within the input complex waveform with peaks correlating to the input ideal power curve .	
<u>MT</u> <u>Measure</u> <u>Rho</u>	Calculates <u>rho</u> given an acquired and ideal waveform. Rho is a measure of correlation between the measured complex waveform and the ideal complex waveform . To generate the ideal waveform, use the digital modulation VIs with the same system parameters given to the digital demodulation VIs, and then modulate the output bit stream returned by the demodulation VI.	

MT Calculate BER VI

Calculates the average <u>bit error rate (BER)</u> against a PN sequence or user-specified reference bit pattern.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Calculate BER After Trigger (Fibonacci PN Sequence)

This polymorphic instance calculates the average BER against a Fibonacci PN sequence. The sequence must be the same as the PN sequence generated by the Fibonacci instance of the <u>MT Generate Bits</u> (poly) VI with a matching PN order.

On first call, or whenever **reset?** is TRUE, the VI steps through each set of N + 1 bits in the input sequence to generate a seed for the PN bit sequence. Subsequent input bits are compared to this PN bit sequence. When the BER on the subsequent input bits is below the specified threshold with the specified confidence, the VI begins reporting the BER. BER calculation begins on the bit following the set of N + 1 bits in which the trigger is found. All subsequent iterations of the VI use a continuation of the same PN sequence until **reset?** is set to TRUE.



input bit stream specifies the bit sequence over which BER is calculated. This bit stream should include only data bits of the PN sequence.

 \mathbf{N}

Note To avoid faulty triggering caused by synchronization and/or guard bits, NI recommends that the **input bit stream** be a phase-continuous PN sequence such as that generated by the <u>MT Generate Bits (poly)</u> VI.

PN sequence order specifies the order of the transmitted PN bit sequence. This parameter must match the PN sequence order of the bits transmitted to the device under test. Valid values are between 5 and 31. The default is 9.

BER trigger threshold specifies the BER threshold that must be exceeded for the **trigger found?** decision. This value determines when the BER measurement begins; bits before this value and seed bits are not included in the BER measurement. Valid values are between 0.0 and 1.0. The default is 0.1.



Note The number of bits examined for the pretrigger BER

used for comparison to the **BER trigger threshold** is determined by the specified **confidence** and BER trigger threshold values. The following images show the range of required input lengths for a **confidence** input in the range [0.5,0.99] and a **BER trigger threshold** input in the range [0.1,0.5].



- **confidence** specifies the level of confidence that the true BER of the bits following a trigger will be below the specified **BER trigger threshold**. Values between 0.0 and 1.0 represent proportional confidence. For example, a value of 0.95 (default) specifies 95% confidence. The default is 0.95 (95% confidence).
 - **Tip** Specify a value of -1 to set the minimum number of bits required for a trigger to the length of the specified **reference bit pattern**. This option is useful when the **input bit stream** is very short.
 - Note The specified **confidence** can only be reached after processing a minimum number of bits in the **input bit stream**. As such, this parameter indirectly specifies the minimum **input bit stream** length in which a trigger can occur. Specify a **confidence** value of -1 to allow triggering on **input bit stream** lengths equal to 3 × (*PN sequence order*).

reset? specifies whether the VI continues bit error rate calculations from a previous iteration. Set **reset?** to TRUE to reset the trigger

and the **accumulated BER** measurement. A **reset?** setting of FALSE maintains the previous synchronization and does not clear the existing **accumulated BER** value. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **BER** returns the measured bit error rate as the ratio of erroneous bits to bits following a trigger in the current **input bit stream**. The number of bits considered for this measurement depends on the conditions shown in the following table.

Scenario	Conditions	BER Output
1.	 reset? set to TRUE or first call, and trigger not detected 	1.0
	in the current	

	input bit stream	
2.	 reset? set to FALSE, and trigger not detected in the current input bit stream, or trigger not detected in previous calls to the same instance of this VI 	1.0
3.	 reset? set to TRUE or first call, and trigger detected in the current input bit stream 	BER calculation based on the length of bits following the set of <i>N</i> + 1 bits in which the trigger was detected, where <i>N</i> is the specified PN sequence order
4.	 reset? set to FALSE, and trigger detected in 	BER calculation based on the length of the input bit stream

|--|

accumulated BER returns the measured bit error rate as the ratio of erroneous bits to bits following a trigger. The number of bits considered for this measurement depends on the conditions in the following table.

Scenario	Conditions	BER Output
1.	 reset? set to TRUE or first call, and trigger not detected in the current input bit stream 	1.0
2.	 reset? set to FALSE, and trigger not detected in the current input bit stream, or trigger not detected in previous calls to the same 	1.0

	instance of this VI	
3.	 reset? set to TRUE or first call, and trigger detected in the current input bit stream 	BER calculation based on the length of bits following the set of <i>N</i> + 1 bits in which the trigger was detected, where <i>N</i> is the specified PN sequence order
4.	 reset? set to FALSE, and trigger detected in previous calls to the same instance of this VI 	Accumulated BER calculation based on the length of the input bit stream .

- **TED** trigger found? returns whether the VI found a trigger. When trigger found? is TRUE, the trigger was found in the input bit stream. When trigger found? is FALSE, the trigger was found in the current or a previous input bit stream.
- **trigger found index (bits)** returns the index value of the first bit where the bit pattern matched the specified PN sequence. This index value indicates the number of bits since the last reset that are not PN data. For example, if the transmitter is off for 30 bits, this parameter returns 30.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu

for more information about the error.

- **Status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Calculate BER After Trigger (PN Sequence)

This polymorphic instance calculates the average BER against a Galois PN sequence. The sequence must be the same as the PN sequence generated by the Galois instance of the <u>MT Generate Bits (poly)</u> VI with a matching PN order. All subsequent iterations of the VI use a continuation of the same PN sequence until **reset?** is set to TRUE.



input bit stream specifies the bit sequence over which BER is calculated. This bit stream should include only data bits of the PN sequence.

 \mathbb{N}

Note To avoid faulty triggering caused by synchronization and/or guard bits, NI recommends that the **input bit stream** be a phase-continuous PN sequence such as that generated by the <u>MT Generate Bits (poly)</u> VI.

PN sequence order specifies the order of the transmitted PN bit sequence. This parameter must match the PN sequence order of the bits transmitted to the device under test. Valid values are between 5 and 31. The default is 9.

BER trigger threshold specifies the BER threshold that must be exceeded for the **trigger found?** decision. This value determines when the BER measurement begins; bits before this value and seed bits are not included in the BER measurement. Valid values are between 0.0 and 1.0. The default is 0.1.

Note The number of bits examined for the pretrigger BER used for comparison to the **BER trigger threshold** is determined by the specified **confidence** and BER trigger threshold values. The following images show the range of required input lengths for a **confidence** input in the range [0.5,0.99] and a **BER trigger threshold** input in the range [0.1,0.5].



confidence specifies the level of confidence that the true BER of the bits following a trigger will be below the specified **BER trigger threshold**. Values between 0.0 and 1.0 represent proportional confidence. For example, a value of 0.95 (default) specifies 95% confidence. The default is 0.95 (95% confidence).

- **Tip** Specify a value of -1 to set the minimum number of bits required for a trigger to the length of the specified **reference bit pattern**. This option is useful when the **input bit stream** is very short.
- Note The specified **confidence** can only be reached after processing a minimum number of bits in the **input bit stream**. As such, this parameter indirectly specifies the minimum **input bit stream** length in which a trigger can occur. Specify a **confidence** value of –1 to allow triggering on **input bit stream** lengths equal to 3 × (*PN sequence order*).
- **reset?** specifies whether the VI continues bit error rate calculations from a previous iteration. Set **reset?** to TRUE to reset the trigger and the **accumulated BER** measurement. A **reset?** setting of FALSE maintains the previous synchronization and does not clear the existing **accumulated BER** value. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other

VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **BER** returns the measured bit error rate as the ratio of erroneous bits to bits following a trigger in the current **input bit stream**. The number of bits considered for this measurement depends on the conditions shown in the following table.

Scenario	Conditions	BER Output
1.	 reset? set to TRUE or first call, and trigger not detected in the current input bit stream 	1.0
2.	 reset? set to FALSE, and trigger not 	1.0

	detected in the current input bit stream , or • trigger not detected in previous calls to the same instance of this VI	
3.	 reset? set to TRUE or first call, and trigger detected in the current input bit stream 	BER calculation based on the length of bits following the set of <i>N</i> + 1 bits in which the trigger was detected, where <i>N</i> is the specified PN sequence order
4.	 reset? set to FALSE, and trigger detected in previous calls to the same instance of this VI 	BER calculation based on the length of the input bit stream

Description accumulated BER returns the measured bit error rate as the ratio

of erroneous bits to bits following a trigger. The number of bits considered for this measurement depends on the conditions in the following table.

Scenario	Conditions	BER Output
1.	 reset? set to TRUE or first call, and trigger not detected in the current input bit stream 	1.0
2.	 reset? set to FALSE, and trigger not detected in the current input bit stream, or trigger not detected in previous calls to the same instance of this VI 	1.0
3.	 reset? set to TRUE or first 	BER calculation based on the length of bits following the set of $N + 1$ bits in which the trigger was detected, where

	call, and • trigger detected in the current input bit stream	<i>N</i> is the specified PN sequence order
4.	 reset? set to FALSE, and trigger detected in previous calls to the same instance of this VI 	Accumulated BER calculation based on the length of the input bit stream .

- trigger found? returns whether the VI found a trigger. When trigger found? is TRUE, the trigger was found in the input bit stream. When trigger found? is FALSE, the trigger was found in the current or a previous input bit stream.
- **trigger found index (bits)** returns the index value of the first bit where the bit pattern matched the specified PN sequence. This index value indicates the number of bits since the last reset that are not PN data. For example, if the transmitter is off for 30 bits, this parameter returns 30.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information

about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Calculate BER After Trigger (User Pattern)

This polymorphic instance calculates the average BER against a userdefined sequence. The sequence input in the **input bit stream** array must be the same as the sequence specified in the **reference bit pattern** control.

On first call, or whenever **reset?** is TRUE, the VI attempts to locate a trigger within the **input bit stream** by comparing it to the **reference bit pattern** under the constraints set by the **confidence** and **BER trigger threshold** parameters. When the BER on the subsequent input bits is below the specified **threshold** over the number of bits specified by **confidence**, the VI begins reporting the BER.



- **input bit stream** specifies the bit sequence over which BER is calculated. This bit stream should include only data bits of the PN sequence.
 - Note To avoid faulty triggering caused by synchronization and/or guard bits, NI recommends that the **input bit stream** be a phase-continuous PN sequence such as that generated by the <u>MT Generate Bits (poly)</u> VI.
- **reference bit pattern** user-defined sequence defines the bits that are compared to the **input bit stream** to calculate BER and accumulated BER. When this function is reset, the starting bit of this sequence is adjusted to find the minimum BER
- **BER trigger threshold** specifies the BER threshold that must be exceeded for the **trigger found?** decision. This value determines when the BER measurement begins; bits before this value and seed bits are not included in the BER measurement. Valid values are between 0.0 and 1.0. The default is 0.1.
 - $\overline{\mathbb{N}}$
- **Note** The number of bits examined for the pretrigger BER used for comparison to the **BER trigger threshold** is determined by the specified **confidence** and BER trigger

threshold values. The following images show the range of required input lengths for a **confidence** input in the range [0.5,0.99] and a **BER trigger threshold** input in the range [0.1,0.5].



- **confidence** specifies the level of confidence that the true BER of the bits following a trigger will be below the specified **BER trigger threshold**. Values between 0.0 and 1.0 represent proportional confidence. For example, a value of 0.95 (default) specifies 95% confidence. The default is 0.95 (95% confidence).
 - **Tip** Specify a value of -1 to set the minimum number of bits required for a trigger to the length of the specified **reference bit pattern**. This option is useful when the **input bit stream** is very short.
 - Note The specified **confidence** can only be reached after processing a minimum number of bits in the **input bit stream**. As such, this parameter indirectly specifies the minimum **input bit stream** length in which a trigger can occur. Specify a **confidence** value of -1 to allow triggering on **input bit stream** lengths equal to 3 × (*PN sequence order*).

reset? specifies whether the VI continues bit error rate calculations from a previous iteration. Set **reset?** to TRUE to reset the trigger and the **accumulated BER** measurement. A **reset?** setting of FALSE maintains the previous synchronization and does not clear

the existing accumulated BER value. The default is TRUE.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **BER** returns the measured bit error rate as the ratio of erroneous bits to bits following a trigger in the current **input bit stream**. The number of bits considered for this measurement depends on the conditions shown in the following table.

1. • reset? set to TRUE or first call, and 1.0 • trigger not detected in the current input bit otroom • trigger not detected in the current input bit otroom	Scenario	Conditions	BER Output
otroom	1.	 reset? set to TRUE or first call, and trigger not detected in the current input bit 	1.0
Suean		stream	

2.	 reset? set to FALSE, and trigger not detected in the current input bit stream, or trigger not detected in previous calls to the same instance of this VI 	1.0
3.	 reset? set to TRUE or first call, and trigger detected in the current input bit stream 	BER calculation based on the length of bits following the set of <i>N</i> + 1 bits in which the trigger was detected, where <i>N</i> is the specified PN sequence order
4.	 reset? set to FALSE, and trigger detected in previous calls to 	BER calculation based on the length of the input bit stream

|--|

accumulated BER returns the measured bit error rate as the ratio of erroneous bits to bits following a trigger. The number of bits considered for this measurement depends on the conditions in the following table.

Scenario	Conditions	BER Output
1.	 reset? set to TRUE or first call, and trigger not detected in the current input bit stream 	1.0
2.	 reset? set to FALSE, and trigger not detected in the current input bit stream, or trigger not detected in previous calls to the same instance of this VI 	1.0

3.	 reset? set to TRUE or first call, and trigger detected in the current input bit stream 	BER calculation based on the length of bits following the set of <i>N</i> + 1 bits in which the trigger was detected, where <i>N</i> is the specified PN sequence order
4.	 reset? set to FALSE, and trigger detected in previous calls to the same instance of this VI 	Accumulated BER calculation based on the length of the input bit stream .

- **TE** trigger found? returns whether the VI found a user-defined sequence synchronization that exceeded the specified **BER trigger** threshold.
- **trigger found index (bits)** returns the index value of the first bit where the bit pattern matched the specified user-defined pattern. This index value indicates the number of bits since the last reset that didn't contribute to the BER calculation did not exceed the value specified by threshold. For example, if the transmitter is off for 30 bits, this parameter returns 30.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE (checkmark)

to indicate a warning or that no error occurred. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

For MT Calculate BER After Trigger (PN Sequence) VI

- Note While stepping through the **input bit stream** one bit at a time, this VI attempts to locate a trigger using the following procedure:
 - 1. Length-checking the **input bit stream** (minus any bits already examined in previous attempts to find a trigger). If the available length is insufficient to meet the specified **confidence**, the VI buffers input bits and exits without locating a trigger.
 - 2. Generating a trial PN sequence seed (trial seed) based on the next N + 1 bits, where N is the expected PN sequence order. Generating a trial PN sequence based on the trial seed, with a length of the specified **confidence**.
 - 3. Calculating a trial BER value using the trial PN sequence as a reference and comparing this reference to the bits following the N + 1 bits that generate the trial seed. The VI returns to step 1 if the trial BER is above the specified **BER trigger threshold**.
 - 4. Generating a second trial PN sequence seed, using the N + 1 bits immediately following the N + 1 bits used to generate the first trial PN sequence seed. Generating a trial PN sequence based on the trial seed with a length of the specified **confidence** (N + 1) bits.
 - 5. Calculating a trial BER value using the trial PN sequence as a reference and comparing this reference to the bits following the N + 1 bits used to generate the trial seed. The VI returns to step 1 if the second trial BER is above the specified **BER trigger threshold**.
 - 6. Returning a **trigger found?** value of TRUE.

BER calculation begins on the bit following the first set of N + 1 bits in which the trigger is found.

The VI is triggered, meaning that the VI only reports BER on bits after a match is found. Hence you do not need to synchronize your transmitter with your unit under test (UUT)—the test can begin before transmission

begins. The **trigger found?** parameter returns whether the BER measurement has started. When **reset?** is set to FALSE, the **trigger found index** parameter returns the starting bit for the block of data in which the trigger initially occurred (this is not necessarily the current block).

On first call, or whenever **reset?** is TRUE, the VI begins a blind search over the **input bit stream** for a **trigger found index** location in which the BER of a number of subsequent bits is below the specified **BER trigger threshold**. The **confidence** parameter specifies the number of bits after the **trigger found index** that are used in the trigger threshold calculation.
MT Measure Burst Timing VI

Locates a burst within the **input complex waveform** with peaks correlating to the input **ideal power curve**.



Note The **upper mask**, **ideal power curve**, and **lower mask** arrays must contain the same number of elements.

Details



input complex waveform specifies the time-aligned and oversampled complex waveform data after matched filtering and frequency offset correction.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies time interval between data points in the acquired **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- **upper mask** specifies the upper mask threshold for burst power in dB. The default is 0.
- **ideal power curve** specifies the best location for the power curve. This curve is correlated against the **input complex waveform** to find the peak correlation. The **ideal power curve** should always be located between the **upper mask** and the **lower mask**. The default is 0.
- **Iower mask** specifies the lower mask threshold for burst power in dB. The default is 0.
- impairment measurement window specifies—relative to the start of the region delineated by the upper mask, ideal power curve, and lower mask parameters—the portion of the input complex

waveform that is measured for amplitude droop and crest factor.

- **start index** specifies the index of the first sample of the measurement window. The default is 0.
- width specifies the number of samples over which amplitude droop and crest factor are measured. This value must be 2 or greater. The default is 100.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **PIE pass?** returns whether the peak-correlated portion of the **input complex waveform** meets the **upper mask** and **lower mask** specifications. If this parameter is TRUE, all samples fall within the region defined by the **upper mask** and **lower mask** parameters.
- **first error sample index** returns the index value of the first sample in the peak-correlated portion of the **input complex waveform** that falls outside of the region defined by the **upper mask** and **lower mask** parameters. This index value is an offset into the **upper mask/lower mask** arrays.
- **correlated burst index** returns the index value of the first sample

in the peak-correlated portion of the **input complex waveform**, in samples. This index value is relative to the start of the **input complex waveform**.

crest factor returns the measured crest factor of the power envelope calculated over the specified **impairment measurement window**. The calculation is defined as the dimensionless quantity

 $[10 \times \log(\max p_n)/\text{Mean } p_n]$

where p_n is the power envelope of the **input complex waveform**.

- **amplitude droop** returns the amplitude droop, in dB, calculated over the specified **impairment measurement window**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

For burst signals, *burst timing* refers to the location of the burst obtained by its correlation against an ideal power curve. In addition, an upper and lower mask are used to test whether the burst signal satisfies mask specifications. The following figure illustrates the ideal power curve, upper mask, and lower mask.



When located, the VI compares the peak-correlated section with the **upper mask** and **lower mask** parameters to validate that the peakcorrelated portion falls within the mask specification. Finally, the VI measures and returns the **amplitude droop** and **crest factor** of the peak-correlated portion of the **input complex waveform**.

MT Measure Quadrature Impairments VI

Measures and reports quadrature impairments over a single userspecified window for each of the supported modulation types. This VI must be called once for each measurement window.

Use the pull-down menu to select an instance of this VI.

Select an instance

•

Details

MT Measure PSK Quadrature Impairments

This polymorphic instance calculates and reports PSK quadrature impairments on a symbol-by-symbol basis at the symbol timing.



PSK system parameters specifies parameter values defining the PSK system. Wire the PSK system parameters cluster from the PSK (M) or PSK (Map) instance of the <u>MT Generate System</u> <u>Parameters</u> VI to this cluster. Do not alter the values.

- samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
- **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of states in the array must be 2^N, where N is the number of bits per symbol.
- differential PSK specifies how the PSK modulation represents symbols. Differential operation is used to implement PSK formats such as differential quadrature PSK (DQPSK) and π/4-DQPSK.

disable (0)	Symbols are represented as constellation points. This is the default value.
enable (1)	Symbols are represented as the transitions between constellation points.

PSK type specifies the type of PSK modulation.

normal	Sets the modulation type to regular PSK. This is the default value.
shifted	Rotates the constellation by π/M each symbol.
offset	Sets the modulation type to offset quadrature phase- shift keying (OQPSK). This modulation scheme is a

form of phase-shift keying in which four different phase angles are used. This scheme is sometimes referred to as staggered quadrature phase-shift keying (SQPSK). For offset PSK, the ideal symbol timing for Q is offset by 1/2 of a symbol period from the ideal symbol timing for I. **offset** is currently only supported for M = 4.

- **recovered complex waveform** specifies the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. Wire the recovered complex waveform parameter of the <u>MT Demodulate</u> <u>PSK</u> VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **input bit stream** specifies the demodulated bit stream from the **output bit stream** parameter of the <u>MT Demodulate PSK</u> VI.
- **impairment measurement window** specifies the window over which impairments are measured.
 - **start index** specifies the index of the first sample of the measurement window. The default is 0.
 - width specifies the number of symbols over which to measure impairments. A value of -1 (default) measures impairments over all symbols. Positive values must be two or greater.
- impairment definition specifies which set of equations is used to represent impairments.

In the equations in the following table, *I* is the real component and *Q* is the imaginary component of each sample in the **input complex waveform**. *I*' and *Q*' are the real and imaginary components of the corresponding sample in the **output complex**

waveform, $I \circ is I DC Offset (\%) / 100$, and $Q \circ is Q DC Offset (\%) / 100$.

Vertical The definition uses the following equations for I/Q **Shear** (0) impairments: *l'* = a * *l* + *l* 。 $Q' = a * \sin(\varphi) * I + b * \cos(\varphi) * Q + Q$ where φ = the specified quadrature skew, in radians y = 10(IQ gain imbalance/20) a = _Y * b $b = \sqrt{\frac{2}{1+\gamma^2}}$ In matrix form, these equations are represented by $\begin{bmatrix} I'\\ O' \end{bmatrix} = S \begin{bmatrix} I\\ Q \end{bmatrix} + \begin{bmatrix} I_\circ\\ Q_\circ \end{bmatrix}$ where $S = \begin{bmatrix} a & 0\\ a * \sin \varphi & b * \cos \varphi \end{bmatrix}$ Axis With this option selected, this VI uses an impairment **Shear** (1) definition that simplifies the conversion between measured impairments and their inverse impairments. For example, you may want to measure the I/Q impairments of a system and compensate for those impairments by applying the inverse impairments to the generated or received waveform. Using the Axis Shear definition, given a measured skew and imbalance (in dB), the inverse impairments are -1.0 * skew and -1.0 **imbalance*. This definition uses the following equations for IQ impairments: $I' = I * \sqrt{\gamma} - Q * \left(\frac{\varphi}{2}\right) + I_{\circ}$ $Q' = -I * \left(\frac{\varphi}{2}\right) + Q * \left(\frac{1}{\sqrt{\gamma}}\right) + Q$

where $r = 10^{(IQ \text{ gain imbalance/20})}$ $\varphi = \text{the specified quadrature skew, in radians}$ In matrix form, these equations are represented by $\begin{bmatrix} I' \\ Q' \end{bmatrix} = S \begin{bmatrix} I \\ Q \end{bmatrix} + \begin{bmatrix} I_* \\ Q_* \end{bmatrix}$ where $S = \begin{bmatrix} \sqrt{\gamma} & -\varphi/2 \\ -\varphi/2 & \frac{1}{\sqrt{\gamma}} \end{bmatrix}$

- **reset?** specifies how the VI handles bits from partial symbols in the **input bit stream**. When **reset?** is set to TRUE, bits making up incomplete symbols are discarded. When **reset?** is set to FALSE, the VI saves the leftover bits and starts with them on the next iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **IQ gain imbalance** returns the measured ratio of I gain to Q gain,

in dB.

Tip For shifted BPSK (also called π/2 BPSK with or without differential encoding), certain measurements, including IQ gain imbalance, quadrature skew, and DC offset measurements, are not performed by the current version of the Modulation Toolkit. The current version of the Modulation Toolkit. The current version of the Modulation Toolkit returns NaN in these cases. Other measurements, specifically modulation error ratio, error vector magnitude, magnitude error, and phase error, are valid and returned for shifted (π/2) BPSK.

phase error returns the measured <u>phase error</u> in degrees. Notice that the phase offset is removed by the demodulator and is excluded from this measurement.

- **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
- **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
- **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
- **individual symbol measurements** returns the impairment value for each individual symbol.
- **EVM** returns the measured <u>error vector magnitude (EVM)</u> expressed as a percentage.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- **modulation error ratio** returns the measured <u>modulation error ratio</u> in dB.
- **magnitude error** returns the measured magnitude error as a

percentage of full scale. Magnitude error is the magnitude difference between the ideal and the actual measured symbol locations.

RMS measurement returns the rms impairment value calculated over the **impairment measurement window**.

- **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
- **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
- **individual symbol measurements** returns the impairment value for each individual symbol.

DC offset measurements returns the measured DC offset of the <u>I</u> or <u>Q</u> waveforms as a percentage of the largest I and Q value in the symbol map of the **recovered complex waveform**.

- **I** returns the DC offset of the I waveform, expressed as a percentage of the largest I or Q value in the symbol map.
- **Q** returns the DC offset of the Q waveform, expressed as a percentage of the largest I or Q value in the symbol map.
- **origin offset** returns the offset, in dB, of the constellation origin from its ideal location.
- **quadrature skew** returns the measured <u>quadrature skew</u> of the complex waveform in degrees.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about

the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Measure QAM Quadrature Impairments

This polymorphic instance calculates and reports <u>QAM</u> quadrature impairments.



recovered complex waveform specifies the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. Wire the recovered complex waveform parameter of the MT Demodulate QAM VI to this parameter.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **input bit stream** specifies the demodulated bit stream from the **output bit stream** parameter of the <u>MT Demodulate QAM</u> VI.
- **impairment measurement window** specifies the window over which impairments are measured.
 - **start index** specifies the index of the first sample of the measurement window. The default is 0.
 - width specifies the number of symbols over which to measure impairments. A value of -1 (default) measures impairments over all symbols. Positive values must be two or greater.
- QAM system parameters specifies parameter values defining the QAM system. Wire the QAM system parameters cluster returned by the QAM (M) or QAM (Map) instance of the <u>MT Generate</u>

System Parameters VI to this cluster. Do not alter the values.

samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.



Note The Modulation Toolkit demodulation and detector VIs use timing recovery, which is optimized for four or more samples per symbol.

symbol map specifies an ordered array that maps each symbol value to its desired coordinates in the complex plane. The number of QAM states in the array must be 2^N, where N is the number of bits per symbol. The length of the vector for the symbol(s) farthest from the origin must be 1.

impairment definition specifies which set of equations is used to represent impairments.

In the equations in the following table, *I* is the real component and *Q* is the imaginary component of each sample in the **input complex waveform**. *I*' and *Q*' are the real and imaginary components of the corresponding sample in the **output complex waveform**, *I* $_{\circ}$ is **I DC Offset (%)** / 100, and *Q* $_{\circ}$ is **Q DC Offset (%)** / 100.

Vertical Shear (0) The definition uses the following equations for I/Q impairments: $l' = a * l + l \circ$ $Q' = a * \sin(\varphi) * l + b * \cos(\varphi) * Q + Q \circ$ where $\varphi =$ the specified quadrature skew, in radians $r = 10^{(IQ \text{ gain imbalance/20})}$ a = r * b $b = \sqrt{\frac{2}{1+\gamma^2}}$ In matrix form, these equations are represented by

$$\begin{bmatrix} I_{Q'}^{I} = s \begin{bmatrix} I_{Q} \end{bmatrix} + \begin{bmatrix} I_{Q} \\ Q_{z} \end{bmatrix} \\ \text{where} \\ s = \begin{bmatrix} a & 0 \\ a * \sin \varphi & b * \cos \varphi \end{bmatrix} \\ \\ \hline \textbf{Axis} \\ \textbf{Shear (1)} \\ \textbf{definition that simplifies the conversion between measured impairments and their inverse impairments. For example, you may want to measure the I/Q impairments of a system and compensate for those impairments by applying the inverse impairments to the generated or received waveform. Using the Axis Shear definition, given a measured skew and imbalance (in dB), the inverse impairments are -1.0 * skew and -1.0 * imbalance. This definition uses the following equations for IQ impairments:
$$I' = I * \sqrt{\gamma} - Q * \left(\frac{\varphi}{2}\right) + I \cdot Q' = -I * \left(\frac{\varphi}{2}\right) + Q * \left(\frac{1}{\sqrt{\gamma}}\right) + Q \cdot Q' = -I * \left(\frac{\varphi}{2}\right) + [Q_{z}] \end{bmatrix} \\ \hline \textbf{where} \\ \gamma = 10^{(IQ gain imbalance/20)} \\ \textbf{\phi} = the specified quadrature skew, in radians In matrix form, these equations are represented by \\ \begin{bmatrix} I_{Q'}' \\ I_{Q'}' \end{bmatrix} = s \begin{bmatrix} I_{Q} \\ I_{Q} \end{bmatrix} + \begin{bmatrix} I_{z} \\ I_{Q'} \\ -\varphi/2 & \frac{1}{\sqrt{\gamma}} \end{bmatrix} \\ \hline \textbf{where} \\ s = \begin{bmatrix} \sqrt{\gamma} & -\varphi/2 \\ -\varphi/2 & \frac{1}{\sqrt{\gamma}} \end{bmatrix} \\ \hline \textbf{where} \\ s = \begin{bmatrix} \sqrt{\gamma} & -\varphi/2 \\ -\varphi/2 & \frac{1}{\sqrt{\gamma}} \end{bmatrix}$$$$

reset? specifies how the VI handles bits from partial symbols in the **input bit stream**. When **reset?** is set to TRUE, bits making up incomplete symbols are discarded. When **reset?** is set to FALSE, the VI saves the leftover bits and starts with them on the next iteration. The default is TRUE.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **IQ gain imbalance** returns the measured ratio of I gain to Q gain, in dB.
- **phase error** returns the measured <u>phase error</u> in degrees. Notice that the phase offset is removed by the demodulator and is excluded from this measurement.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- **EVM** returns the measured <u>error vector magnitude (EVM)</u> expressed as a percentage.
 - **PIEL RMS measurement** returns the rms impairment value

calculated over the **impairment measurement window**. **peak measurement** returns the peak impairment value measured over the impairment measurement window. peak symbol index returns the index of the symbol having the peak magnitude of impairment. **individual symbol measurements** returns the impairment value for each individual symbol. modulation error ratio returns the measured modulation error ratio in dB. **magnitude error** returns the measured magnitude error as a percentage of full scale. Magnitude error is the magnitude difference between the ideal and the actual measured symbol locations. **RMS measurement** returns the rms impairment value calculated over the impairment measurement window. **peak measurement** returns the peak impairment value measured over the impairment measurement window. **peak symbol index** returns the index of the symbol having the peak magnitude of impairment. **individual symbol measurements** returns the impairment value for each individual symbol. **DC offset measurements** returns the measured DC offset of the I or Q waveforms as a percentage of the largest I and Q value in the symbol map of the recovered complex waveform. **I** returns the DC offset of the I waveform, expressed as a percentage of the largest I or Q value in the symbol map. Q returns the DC offset of the Q waveform, expressed as a percentage of the largest I or Q value in the symbol map. **origin offset** returns the offset, in dB, of the constellation origin from its ideal location. **pull quadrature skew** returns the measured quadrature skew of the complex waveform in degrees. EVALUATE: EVALUATE REPORT OF WARNING INFORMATION OUT OF A VI to be used

by other VIs. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Measure MSK Quadrature Impairments

This polymorphic instance calculates <u>MSK</u> quadrature impairments on a point-by-point basis on the oversampled waveform.



recovered complex waveform specifies the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. Wire the recovered complex waveform parameter of the MT Demodulate MSK VI to this parameter.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- detected complex waveform specifies the ideal oversampled waveform. Wire the detected complex waveform parameter of the <u>MT Demodulate MSK</u> VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - **Y** specifies the ideal oversampled waveform as a complex-valued array.
- **impairment measurement window** specifies the window over which impairments are measured.
 - **start index** specifies the index of the first sample of the

measurement window. The default is 0.

width specifies the number of symbols over which to measure impairments. A value of -1 (default) measures impairments over all symbols. Positive values must be two or greater.

samples per symbol specifies the number of samples per symbol in the modulated complex waveform. The default is 16.

impairment definition specifies which set of equations is used to represent impairments.

In the equations in the following table, *I* is the real component and *Q* is the imaginary component of each sample in the **input complex waveform**. *I*' and *Q*' are the real and imaginary components of the corresponding sample in the **output complex waveform**, $I \circ$ is **I DC Offset (%)** / 100, and $Q \circ$ is **Q DC Offset (%)** / 100.

Vertical The definition uses the following equations for I/Q **Shear** (0) impairments: $l' = a * l + l_{\circ}$ $Q' = a * \sin(\varphi) * I + b * \cos(\varphi) * Q + Q$. where φ = the specified quadrature skew, in radians r = 10(IQ gain imbalance/20) $a = \gamma * b$ $b = \sqrt{\frac{2}{1+\gamma^2}}$ In matrix form, these equations are represented by $\begin{bmatrix} I'\\ O' \end{bmatrix} = S \begin{bmatrix} I\\ Q \end{bmatrix} + \begin{bmatrix} I_\circ\\ Q_\circ \end{bmatrix}$ where $S = \begin{bmatrix} a & 0\\ a * \sin \varphi & b * \cos \varphi \end{bmatrix}$ With this option selected, this VI uses an impairment Axis

Shear (1) definition that simplifies the conversion between measured impairments and their inverse impairments. For example, you may want to measure the I/Q impairments of a system and compensate for those impairments by applying the inverse impairments to the generated or received waveform. Using the Axis Shear definition, given a measured skew and imbalance (in dB), the inverse impairments are -1.0 * skew and -1.0 * imbalance. This definition uses the following equations for IQ impairments:

$$I' = I * \sqrt{\gamma} - Q * \left(\frac{1}{2}\right) + I$$
$$Q' = -I * \left(\frac{\varphi}{2}\right) + Q * \left(\frac{1}{\sqrt{\gamma}}\right) + Q$$

where

 $y = 10^{(IQ \text{ gain imbalance/20})}$

 ϕ = the specified quadrature skew, in radians

In matrix form, these equations are represented by

$$\begin{bmatrix} I'\\Q'\end{bmatrix} = S\begin{bmatrix} I\\Q\end{bmatrix} + \begin{bmatrix} I_\circ\\Q_\circ\end{bmatrix}$$

where

$$S = \begin{bmatrix} \sqrt{\gamma} & -\varphi/2 \\ -\varphi/2 & \frac{1}{\sqrt{\gamma}} \end{bmatrix}$$

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

code identifies the error or warning code. Right-click the front

panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **IQ gain imbalance** returns the measured ratio of I gain to Q gain, in dB.
- **phase error** returns the measured <u>phase error</u> in degrees. Notice that the phase offset is removed by the demodulator and is excluded from this measurement.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- **EVM** returns the measured <u>error vector magnitude (EVM)</u> expressed as a percentage.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- modulation error ratio returns the measured modulation error ratio in dB.
- **magnitude error** returns the measured magnitude error as a

percentage of full scale. Magnitude error is the magnitude difference between the ideal and the actual measured symbol locations.

RMS measurement returns the rms impairment value calculated over the **impairment measurement window**.

- **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
- **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
- **individual symbol measurements** returns the impairment value for each individual symbol.

DC offset measurements returns the measured DC offset of the <u>I</u> or <u>Q</u> waveforms as a percentage of the largest I and Q value in the symbol map of the **recovered complex waveform**.

- **I** returns the DC offset of the I waveform, expressed as a percentage of the largest I or Q value in the symbol map.
- **Q** returns the DC offset of the Q waveform, expressed as a percentage of the largest I or Q value in the symbol map.
- **origin offset** returns the offset, in dB, of the constellation origin from its ideal location.
- **quadrature skew** returns the measured <u>quadrature skew</u> of the complex waveform in degrees.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about

the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Measure PAM Quadrature Impairments

This polymorphic instance calculates and reports PAM quadrature impairments on a symbol-by-symbol basis at the symbol timing.



Note Certain measurements like I/Q gain imbalance and quadrature skew are not applicable for the PAM modulation format because of the inherently one-dimensional nature of the PAM constellation.



- **recovered complex waveform** specifies the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. Wire the **recovered complex waveform** parameter of the <u>MT Demodulate</u> <u>PAM</u> VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **input bit stream** specifies the demodulated bit stream from the **output bit stream** parameter of the <u>MT Demodulate PAM</u> VI.
- **impairment measurement window** specifies the window over which impairments are measured.
 - **start index** specifies the index of the first sample of the measurement window. The default is 0.
 - width specifies the number of symbols over which to measure impairments. A value of -1 (default) measures impairments over all symbols. Positive values must be two or greater.

- **PAM system parameters** specifies parameter values defining the PAM system. Wire the **PAM system parameters** cluster returned by either the PAM (M) or PAM (Map) instance of the <u>MT Generate</u> <u>System Parameters</u> VI to this cluster.
 - **samples per symbol** specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each Boolean symbol to its desired coordinates in the complex plane. The number of states in the array must be 2^{*N*}, where *N* is the number of bits per symbol.
- reset? specifies how the VI handles bits from partial symbols in the input bit stream. When reset? is set to TRUE, bits making up incomplete symbols are discarded. When reset? is set to FALSE, the VI saves the leftover bits and starts with them on the next iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

phase error returns the measured <u>phase error</u> in degrees. Notice that the phase offset is removed by the demodulator and is excluded from this measurement.

- **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
- **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
- **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
- **individual symbol measurements** returns the impairment value for each individual symbol.
- **EVM** returns the measured <u>error vector magnitude (EVM)</u> expressed as a percentage.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- **modulation error ratio** returns the measured <u>modulation error ratio</u> in dB.

magnitude error returns the measured magnitude error as a percentage of full scale. Magnitude error is the magnitude difference between the ideal and the actual measured symbol locations.

- **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
- **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
- **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.

[DBL]

individual symbol measurements returns the impairment value for each individual symbol.

- **DC offset measurements** returns the measured DC offset of the or Q waveforms as a percentage of the largest I and Q value in the symbol map of the **recovered complex waveform**.
 - I returns the DC offset of the I waveform, expressed as a percentage of the largest I or Q value in the symbol map.
 - **Q** returns the DC offset of the Q waveform, expressed as a percentage of the largest I or Q value in the symbol map.
 - **origin offset** returns the offset, in dB, of the constellation origin from its ideal location.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Measure ASK Quadrature Impairments

This polymorphic instance calculates and reports ASK quadrature impairments on a symbol-by-symbol basis at the symbol timing.



Note Certain measurements like I/Q gain imbalance and quadrature skew are not applicable for the ASK modulation format because of the inherently one-dimensional nature of the ASK constellation.



- recovered complex waveform specifies the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. Wire the output complex waveform parameter of the <u>MT Demodulate ASK</u> VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **input bit stream** specifies the demodulated bit stream from the **output bit stream** parameter of the <u>MT Demodulate ASK</u> VI.
- **impairment measurement window** specifies the window over which impairments are measured.
 - **start index** specifies the index of the first sample of the measurement window. The default is 0.
 - width specifies the number of symbols over which to measure impairments. A value of -1 (default) measures impairments over all symbols. Positive values must be two or greater.

- ASK system parameters specifies parameter values defining the ASK system. Wire the ASK system parameters cluster returned by the ASK instance of the MT Generate System Parameters VI to this cluster. Do not alter the values.
 - samples per symbol specifies an even, positive number of samples dedicated to each symbol. Multiply this value by the symbol rate to determine the sample rate. The default is 16.
 - **symbol map** specifies an ordered array that maps each symbol to its desired level. The number of ASK levels in the array is 2^{*N*}, where *N* is the number of bits per symbol. The length of the vector for the symbols farthest from the origin is 1.
- reset? specifies how the VI handles bits from partial symbols in the input bit stream. When reset? is set to TRUE, bits making up incomplete symbols are discarded. When reset? is set to FALSE, the VI saves the leftover bits and starts with them on the next iteration. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **phase error** returns the measured <u>phase error</u> in degrees. Notice that the phase offset is removed by the demodulator and is excluded from this measurement.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- **EVM** returns the measured <u>error vector magnitude (EVM)</u> expressed as a percentage.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- **modulation error ratio** returns the measured <u>modulation error ratio</u> in dB.
- magnitude error returns the measured magnitude error as a percentage of full scale. Magnitude error is the magnitude difference between the ideal and the actual measured symbol locations.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.

individual symbol measurements returns the impairment value for each individual symbol.

DC offset measurements returns the measured DC offset of the or Q waveforms as a percentage of the largest I and Q value in the symbol map of the **recovered complex waveform**.

- I returns the DC offset of the I waveform, expressed as a percentage of the largest I or Q value in the symbol map.
- **Q** returns the DC offset of the Q waveform, expressed as a percentage of the largest I or Q value in the symbol map.
- **origin offset** returns the offset, in dB, of the constellation origin from its ideal location.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Measure CPM Quadrature Impairments

Measures and reports quadrature impairments over a single userspecified window. This VI must be called once for each measurement window. This polymorphic instance calculates CPM quadrature impairments on a point-by-point basis on the oversampled waveform.



- recovered complex waveform specifies the time-aligned and oversampled complex waveform data after matched filtering, frequency offset correction, and phase offset correction. Wire the recovered complex waveform parameter of the MT Demodulate CPM VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- detected complex waveform specifies the ideal oversampled waveform. Wire the detected complex waveform parameter of the <u>MT Demodulate CPM</u> VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.

- **impairment measurement window** specifies the window over which impairments are measured.
 - **start index** specifies the index of the first sample of the measurement window. The default is 0.
 - width specifies the number of symbols over which to measure impairments. A value of -1 (default) measures impairments over all symbols. Positive values must be two or greater.
- **samples per symbol** specifies the number of samples per symbol in the modulated complex waveform. The default is 16.
- impairment definition specifies which set of equations is used to represent impairments.

In the equations in the following table, *I* is the real component and *Q* is the imaginary component of each sample in the **input complex waveform**. *I*' and *Q*' are the real and imaginary components of the corresponding sample in the **output complex waveform**, *I* \circ is **I DC Offset (%)** / 100, and *Q* \circ is **Q DC Offset (%)** / 100.

Vertical Shear (0) The definition uses the following equations for I/Q impairments: $l' = a * l + l \circ$ $Q' = a * \sin(\varphi) * l + b * \cos(\varphi) * Q + Q \circ$ where $\varphi =$ the specified quadrature skew, in radians $r = 10^{(IQ \text{ gain imbalance/20})}$ a = r * b $b = \sqrt{\frac{2}{1+\gamma^2}}$ In matrix form, these equations are represented by $\begin{bmatrix} l' \\ Q' \end{bmatrix} = s \begin{bmatrix} l \\ Q \end{bmatrix} + \begin{bmatrix} l \\ Q_* \end{bmatrix}$ where

	$S = \begin{bmatrix} a & 0\\ a * \sin \varphi & b * \cos \varphi \end{bmatrix}$
Axis Shear (1)	With this option selected, this VI uses an impairment definition that simplifies the conversion between measured impairments and their inverse impairments. For example, you may want to measure the I/Q impairments of a system and compensate for those impairments by applying the inverse impairments to the generated or received waveform. Using the Axis Shear definition, given a measured skew and imbalance (in dB), the inverse impairments are $-1.0 * skew$ and $-1.0 * imbalance$. This definition uses the following equations for IQ impairments:
	$I' = I * \sqrt{\gamma} - Q * \left(\frac{\varphi}{2}\right) + I_{\circ}$
	$Q' = -I * \left(\frac{\varphi}{2}\right) + Q * \left(\frac{1}{\sqrt{\gamma}}\right) + Q.$
	where
	$r = 10^{(IQ \text{ gain imbalance/20})}$
	ϕ = the specified quadrature skew, in radians
	In matrix form, these equations are represented by
	$\begin{bmatrix} I'\\Q'\end{bmatrix} = S\begin{bmatrix} I\\Q\end{bmatrix} + \begin{bmatrix} I_\circ\\Q_\circ\end{bmatrix}$
	where
	$S = \begin{bmatrix} \sqrt{\gamma} & -\varphi/2 \\ -\varphi/2 & \frac{1}{\sqrt{\gamma}} \end{bmatrix}$

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain**

Error or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **IQ gain imbalance** returns the measured ratio of I gain to Q gain, in dB.
- **phase error** returns the measured <u>phase error</u> in degrees. Notice that the phase offset is removed by the demodulator and is excluded from this measurement.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- **EVM** returns the measured <u>error vector magnitude (EVM)</u> expressed as a percentage.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.

DBL
modulation error ratio returns the measured <u>modulation error ratio</u> in dB.

- magnitude error returns the measured magnitude error as a percentage of full scale. Magnitude error is the magnitude difference between the ideal and the actual measured symbol locations.
 - **RMS measurement** returns the rms impairment value calculated over the **impairment measurement window**.
 - **peak measurement** returns the peak impairment value measured over the **impairment measurement window**.
 - **peak symbol index** returns the index of the symbol having the peak magnitude of impairment.
 - **individual symbol measurements** returns the impairment value for each individual symbol.
- DC offset measurements returns the measured DC offset of the or Q waveforms as a percentage of the largest I and Q value in the symbol map of the recovered complex waveform.
 - **I** returns the DC offset of the I waveform, expressed as a percentage of the largest I or Q value in the symbol map.
 - **Q** returns the DC offset of the Q waveform, expressed as a percentage of the largest I or Q value in the symbol map.
 - **origin offset** returns the offset, in dB, of the constellation origin from its ideal location.
- **quadrature skew** returns the measured <u>quadrature skew</u> of the complex waveform in degrees.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

For more information about the Vertical Shear definition, refer to Kafadar, Karen. "Statistical Calibration of a Vector Demodulator," HP Journal: June 1988.

MT Measure Rho VI

Calculates <u>rho</u> given an acquired and ideal waveform. Rho is a measure of correlation between the **measured complex waveform** and the **ideal complex waveform**. To generate the ideal waveform, use the digital modulation VIs with the same system parameters given to the digital demodulation VIs, and then modulate the **output bit stream** returned by the demodulation VI.



- **received complex waveform** specifies the actual output waveform from the digital <u>demodulation</u> VI. Wire the **recovered complex waveform** parameter from the digital demodulation VI to this parameter.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- ideal complex waveform specifies the reference waveform. Obtain this waveform by remodulating the **output bit stream** from the digital demodulation VI.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **rho** returns the measured rho of the **received complex waveform**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Impairments

Owning Palette: Digital

Use the VIs on the **Impairments** palette to add several types of baseband impairments to digital signals.

Palette Object	Description
<u>MT Add</u> <u>AWGN</u>	Generates zero-mean complex additive white Gaussian noise (AWGN) with uniform power spectral density and adds it to the complex <u>baseband</u> <u>modulated</u> waveform.
<u>MT Add</u> <u>Phase</u> <u>Noise</u>	Models and applies common sources of <u>phase noise</u> present in communications systems components. This VI creates phase noise by first generating white noise and then applying a 1/ <i>f</i> (inverse <i>f</i>) filter to specify the noise shape. The output of the filter is scaled such that the generated noise density is equal to the specified offset frequency. The phase component of the input signal is then modulated by the filtered noise.
MT Apply IQ Impairments	Applies various I/Q impairments to the complex <u>baseband</u> modulated waveform.
<u>MT Add</u> <u>MultiTone</u>	Generates a multitone waveform and adds it to the complex <u>baseband modulated</u> waveform, returning a signal-plus-tones waveform. You can use the tones-added signal to test transmitter interference rejection.
MT Generate Fading Profile	Generates a user-specified Rayleigh or Rician fading profile for the complex baseband waveform. Call the <u>MT</u> <u>Apply Fading Profile</u> VI to apply the generated profile to a baseband I/Q waveform. You can use this VI for modeling <u>Rayleigh</u> and <u>Rician</u> fading profile distributions using Gans and Jakes models for characterizing flat-fading channels and time-varying frequency selective-fading channels.
<u>MT Apply</u> Fading Profile	Applies a fading profile to fade the complex <u>baseband</u> <u>modulated</u> waveform. The faded waveform can be used to test receiver immunity to fading channels.

MT Add AWGN VI

Generates zero-mean complex additive white Gaussian noise (AWGN) with uniform power spectral density and adds it to the complex <u>baseband</u> <u>modulated</u> waveform.

This VI returns a signal–plus–noise waveform with a user-specified E_b/N_0 , where E_b represents the energy per bit, and N_0 represents the noise variance.

input complex waveform samples per symbol (16) bits per symbol (1) Eb/N0 (10 dB) error in (no error) seed in (0xFFFFFFF) reset? (T)	output complex waveform
reset: (1)	

input complex waveform specifies the modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- samples per symbol specifies the number of samples per information symbol in the input complex waveform. The default is 16.
- **bits per symbol** specifies the number of bits per symbol in the modulation format underlying the **input complex waveform**. For example, the number of bits per symbol in quadrature-phase shift keying (QPSK) is 2. The default is 1.
- **Eb/N0** specifies the desired E_b/N_0 of the **output complex** waveform in dB. The default is 10.0 dB.
- **seed in** specifies (when **reset?** is TRUE) how the noise sample generator is reseeded. When this parameter value is >0, the AWGN generator is reseeded using the seed you specify. Set this value to

-1 to reseed the AWGN generator with a random seed. The default is 0xFFFFFFF (-1).



Note The **seed in** parameter is disregarded when **reset?** is set to FALSE.

- **reset?** specifies whether the VI continues applying AWGN using the previous iteration states. **reset?** must be set to TRUE when you want to reseed the AWGN generator. The AWGN generator maintains its previous state when **reset?** is set to FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the signal-plus-noise complex baseband waveform data.
 - **ID** returns the trigger (start) time of the **Y** array.
 - **DEL** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (]) and

quadrature-phase (Q) data, respectively.

- **seed out** returns a seed for use as **seed in** during the next call to this VI when **reset?** is set to FALSE.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Add MultiTone VI

Generates a multitone waveform and adds it to the complex <u>baseband</u> <u>modulated</u> waveform, returning a signal-plus-tones waveform. You can use the tones-added signal to test transmitter interference rejection.



Note The amplitude of the multitone waveform is the sum of the specified **amplitude** and **reference power** values.

input complex waveform
reference power (dBm)
error in (no error)
reset? (T)

- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (]) and quadrature-phase (Q) data, respectively.
- tones specifies the frequency and amplitude of tones to be generated.
 - **freq** specifies tone frequency, in hertz. The default is 0.0 Hz.
 - **amplitude** specifies tone amplitude, in dB, relative to the specified **reference power**. The default is 0.0 dB.
- **reference power** specifies the reference power for the **amplitude** parameter in dBm. The default is 0.0 dBm.
- **reset?** specifies whether the VI continues tone generation using the previous iteration states. **reset?** must be set to TRUE whenever you want to reset the phase of the generated multitone to 0. Set this parameter to FALSE to continue the generated multitone phase from the previous iteration. The default is TRUE.
- error in (no error) can accept error information wired from

previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the signal-plus-tones complex baseband waveform data.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **besilon dt** returns the time between values in the **Y** array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Add Phase Noise VI

Models and applies common sources of <u>phase noise</u> present in communications systems components. This VI creates phase noise by first generating white noise and then applying a 1/*f* (inverse *f*) filter to specify the noise shape. The output of the filter is scaled such that the generated noise density is equal to the specified offset frequency. The phase component of the input signal is then modulated by the filtered noise.

<u>Details</u>



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (<u>I</u>) and quadrature-phase (<u>Q</u>) data, respectively.
- **Offset frequency** specifies the offset frequency in hertz for the specified **noise density**. This is the frequency at which the 1 Hz bandwidth spectral density of the ideal inverse-f noise waveform is equal to specified **noise density**. Valid entries for this parameter must be in the range 0.7 mHz to $f_s/2$, where f_s is the sampling frequency (1/dt) of the **input complex waveform**. The default is 200,000 Hz (200 kHz).
- noise density specifies the single sideband noise contribution in a 1 Hz bandwidth relative to the total power at the specified offset frequency (Hz). The default is –120 dBc/Hz.

inverse f exponent for noise shape specifies the exponent of the desired 1/*f* curve. The inverse *f* curve specifies the spectral shape of the <u>phase noise</u>. Specify a value in the range –3.0 to 3.0. The default is 2.

- seed in specifies a seed for the noise sample generator. Set this value to -1 (xFFFFFFF) to reseed the noise generator with a random seed. The random signal generator is reseeded on first call to this VI and when reset? is set to TRUE. When reset? is FALSE this parameter is ignored on subsequent calls. The default is -1 (xFFFFFFF).
- **TEN** reset? specifies whether the VI reuses the noise shaping filter and reseeds the noise generator on subsequent calls. When reset? is TRUE (default), the VI designs/redesigns noise-shaping IIR filter and reseeds the noise generator using the seed in parameter value. When reset? is FALSE, the VI uses the previously designed IIR filter, and reseeds the noise generator with the seed out parameter, upon each subsequent call. The VI always resets on first call. The default is TRUE.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **output complex waveform** returns the signal-plus-noise complex baseband waveform data.
 - **t0** returns the trigger (start) time of the **Y** array.
 - **bell** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **seed out** returns the seed used in the subsequent call to this VI when **reset?** set to FALSE.
 - **Tip seed out** is useful when applying phase noise that has a shape including multiple offset frequency regions. In these cases, call this VI multiple times in series with the **seed out** parameter from the previous call wired to the **seed in** parameter for the next call. Doing so reduces the possibility of correlation of the noise applied by subsequent calls.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

- Note Multiregion inverse *f* rolloff noise shapes can be implemented by calling the MT Add Phase Noise VI multiple times in series on a signal, with each call defining the inverse *f* rolloff and desired level at a specific offset frequency. When combining individual MT Add Phase Noise VIs in this way, refer to the following considerations:
 - Wire the **seed out** parameter from a previous call to the **seed in** parameter of a subsequent call to reduce the likelihood of correlation between the individual noise sources.
 - The overall noise spectrum shape in a particular offset frequency region is the sum of contributions from the individual noise sources in that region. As such, at any particular offset frequency, a noise contribution from one source can overwhelm the contribution from another source.

MT Apply Fading Profile VI

Applies a fading profile to fade the complex <u>baseband modulated</u> waveform. The faded waveform can be used to test receiver immunity to fading channels.

Use the pull-down menu to select an instance of this VI.

Select an instance

•

Details

MT Apply Selective Fading Profile

Applies a multipath fading profile to the input complex waveform.



input complex waveform specifies the modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **fading profile** specifies a two-dimensional array of sample-bysample profiles to be applied to the **input complex waveform** for each path in the multipath channel. At each call, the VI begins applying this **fading profile** from the index point where it left off on the previous iteration unless **reset?** is set to TRUE. Wire the **fading profile** parameter of the <u>MT Generate Fading Profile</u> VI to this parameter.
- **power delay profile** specifies the arrival time, in seconds, of the different ray paths versus their received power, in dB. The times are relative to the arrival of the first ray path. The first column of the power delay profile array should be the relative power loss (in dB), and the second column represents the relative time of arrival (in seconds).
- **reset?** specifies whether the VI begins applying the **fading profile** at the index point where it left off on the last iteration. Set **reset?** to TRUE if you want to begin applying the **fading profile** at index point 0. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any

functionality should be bypassed in the event of errors from other VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the faded modulated complex baseband waveform data. This parameter is always the same size as the **input complex waveform**, regardless of the size of the **fading profile**.
 - **ID** returns the time of first value in **Y** array.
 - **besilt** returns the time between values in the **Y** array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for

more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Apply Flat Fading Profile

Applies a flat fading profile to the input complex waveform.

 $\overline{\mathbb{N}}$

Note This instance normalizes the **fading profile** parameter to ensure that the power in the **input complex waveform** is equal to the fading variance. The Modulation Toolkit 4.0 version of MT Apply Flat Fading does not apply this normalization algorithm.



input complex waveform specifies the modulated complex baseband waveform data.

- **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
- **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
- Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **fading profile** specifies the sample-by-sample profile to be applied to the **input complex waveform**. At each call, the VI begins applying this **fading profile** from the index point where it left off on the previous iteration unless **reset?** is set to TRUE. Wire the **fading profile** parameter of the <u>MT Generate Fading Profile</u> VI to this parameter.
- **reset?** specifies whether the VI begins applying the **fading profile** at the index point where it left off on the last iteration. Set **reset?** to TRUE if you want to begin applying the **fading profile** at index point 0. The default is TRUE.

error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **output complex waveform** returns the faded modulated complex baseband waveform data. This parameter is always the same size as the **input complex waveform**, regardless of the size of the **fading profile**.
 - **ID** returns the time of first value in **Y** array.
 - **bell** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain**

Warning from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

If the **fading profile** is shorter than the **input complex waveform**, the VI repeats the **fading profile** until the entire **input complex waveform** is covered. If the **fading profile** is longer than the **input complex waveform**, the VI begins applying the **fading profile** at the index point where it left off during the previous iteration (when **reset?** is set to FALSE). However, to maintain the phase continuity of the **input complex waveform**, the **profile length** should be equal to the input waveform length.

Normalizing the Rician Fading Profile

To match the **output complex waveform** power to the specified **fading variance**, this VI first normalizes the power array of the **power delay profile**, and the amplitude corresponding to power element is then multiplied by the corresponding **fading profile**, f_{kl} , where l = 1, 2, ..., L, and *L* is the **profile length**.

For the selective Rayleigh profile, the power coefficients, C_k^2 , corresponding to each path are shown in the following equation:

 $C_k^2 = 10^{P_k/10}$

The normalized power coefficients can then be represented by

$$\overline{C}_k^2 = \frac{C_k^2}{\sum_{k=1}^N C_k^2}$$

for *k* = 1,...,*N*.

For the selective Rician profile, the power in the line-of-sight (LOS) path is given by

 $P_{LOS} = (k+1) * P_1$

The normalized power coefficients can then be represented by the following equation:

 $\overline{C}_{k}^{2} = \frac{C_{k}^{2}}{(k_{LOS} + 1) * C_{1}^{2} + \sum_{k=2}^{N} C_{k}^{2}}$

By multiplying these power coefficients with the generated fading profile, f_{kl} , for each path, we can obtain the amplitude, a_{kl} , to apply to the Rician profile as shown in the following equation:

$$a_{kl} = \overline{C}_k * f_{kl}$$

for k = 1,...,N and l = 1,...,L, where L is the **profile length**.

The time array, τ_k , is approximated to an integer multiple of the sampling duration, *dt*, to obtain the integer delay, n_k . The integer delay is applied to the **input complex waveform** as shown by the following equation:

$$n_k = \frac{\tau_k}{dt}$$

The **input complex waveform** is then delayed for each path by $y_k[n] = x[n - n_k]$, where n = 1,...,M and *M* is the **input complex waveform** size.

Finally, $y_k[n]$ is point-by-point multiplied with amplitude coefficients to obtain the **output complex waveform** as shown by the following equation:

$$y[n] = \sum_{k=1}^{N} a_{kl} * x[n - n_k]$$

In the MT Apply Selective Fading Profile VI, the power from the **power delay profile**, specified for each path, is applied to the corresponding **fading profile** generated by the <u>MT Generate Fading Profile</u> VI. The **power delay profile** is specified in terms of power, P_k , and time, τ_k , for each path, where k = 1,2...,N. The delay in the **power delay profile** is approximated to an integer multiple of the sampling duration, τ_k/dt for each path, for k = 1,...N, where N is the total number of paths. The input complex waveform is delayed by this amount and then multiplied by the corresponding attenuated fading profile. All these paths are summed to calculate the received signal, y(t), as illustrated by the following figure.



References

Rappaport, Theodore S. *Wireless Communications: Principles and Practice (2nd Edition).* Prentice Hall PTR, December 2001.

Trantner, William H., and Theodore S. Rappaport.*Principles of Communication Systems Simulation with Wireless Applications.* Prentice Hall PTR, January 2004.

Dent, P. et al. "Jakes Fading Model Revisited," *Electronics Letters*: June 1993.

MT Apply IQ Impairments VI

Applies various I/Q impairments to the complex <u>baseband</u> <u>modulated</u> waveform.

Details



- **input complex waveform** specifies the modulated complex baseband waveform data.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- **Q DC offset** specifies the desired DC offset of the Q waveform as a percentage of full scale of the **input complex waveform**. Valid values are between –100 and +100. The default is 0.0.
- **IQ gain imbalance** specifies the desired ratio of I gain to Q gain, in dB. Valid values are between –6.0 and +6.0. The default is 0.0.
- **quadrature skew** specifies the desired quadrature skew of the complex waveform in degrees. Valid values are between –30.0 and +30.0 degrees. The default is 0.0.
- I DC offset specifies the desired DC offset of the I waveform as a percentage of full scale of the input complex waveform. Valid values are between −100 and +100. The default is 0.0.
- impairment definition specifies which set of equations is used to represent impairments.

In the equations in the following table, *I* is the real component and *Q* is the imaginary component of each sample in the **input complex waveform**. *I*' and *Q*' are the real and imaginary components of the corresponding sample in the **output complex waveform**, *I* $_{\circ}$ is **I DC Offset (%)** / 100, and *Q* $_{\circ}$ is **Q DC Offset (%)** / 100.

 $I' = I * \sqrt{\gamma} - Q * \left(\frac{\varphi}{2}\right) + I_{*}$ $Q' = -I * \left(\frac{\varphi}{2}\right) + Q * \left(\frac{1}{\sqrt{\gamma}}\right) + Q_{*}$ where $\gamma = 10^{(IQ \text{ gain imbalance/20})}$ $\varphi = \text{the specified quadrature skew, in radians}$ In matrix form, these equations are represented by $\begin{bmatrix}I'\\Q'\end{bmatrix} = S\begin{bmatrix}I\\Q\end{bmatrix} + \begin{bmatrix}I_{*}\\Q_{*}\end{bmatrix}$ where $S = \begin{bmatrix}\sqrt{\gamma} & -\varphi/2\\ -\varphi/2 & \frac{1}{\sqrt{\gamma}}\end{bmatrix}$

- **frequency offset** specifies the desired frequency offset to apply to the **output complex waveform**, in hertz. The default is 0.0.
- **reset?** specifies whether the VI continues applying the impairments using the previous iteration states. **reset?** must be set to TRUE when you want to restart the impairment generation process. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

abc

source describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **output complex waveform** returns the impaired complex baseband modulated waveform data.
 - **IDENTIFY and SET UP: IDENTIFY and SET UP:**
 - **DEED IN THE INTER SET UP:** IN THE INTER SET IN THE INTERNATION I
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

For more information about the Vertical Shear definition, refer to Kafadar, Karen. "Statistical Calibration of a Vector Demodulator," HP Journal: June 1988.

Visualization

Owning Palette: Digital

Use the VIs on the **Visualization** palette to create eye and constellation diagrams, IQ-to-XY graphs, and trellis diagrams.

Palette Object	Description
MT Format Constellation	Prepares a signal for presentation on a graph showing the detected symbol locations and the transitions between those symbols. The VI specifies a complex-valued waveform and displays a constellation plot of the waveform contents.
<u>MT Format</u> Eye Diagram	Specifies a complex- or real-valued waveform, divides it into segments, and displays those segments as plots on a waveform graph. This VI determines the segment length based on the symbol rate and eye length parameters.
<u>MT Display</u> <u>3DEye</u>	Divides an input complex-valued waveform into segments and displays those segments as overlapping plots on a 3D graph. The VI determines the segment length based on the symbol rate and eye length parameters.
<u>MT Format</u> <u>Trellis</u> Diagram	Receives a complex-valued waveform, divides it into segments of (trellis length / symbol rate), and displays those segments as a trellis diagram on a standard LabVIEW waveform graph referenced by the trellis reference parameter.

MT Display 3DEye VI

Divides an input complex-valued waveform into segments and displays those segments as overlapping plots on a 3D graph. The VI determines the segment length based on the **symbol rate** and **eye length** parameters.

<u>Details</u>



- waveform specifies the modulated data for plotting as a 3D eye diagram.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbol rate** specifies the symbol rate of the modulated data, in hertz. The default is 1.0.
- eye length specifies the number of symbol periods on the horizontal (X) scale of the 3D eye diagram. The default is 1.
- **color** specifies the 24-bit color in which to plot the segments. If **color** > white (white = $2^{24} 1$), then each segment is plotted in a different random color. The default is GREEN 255.
- optional settings specifies optional settings that control the plot update rate and enable programmatic interruption. This parameter can be safely left unwired.
 - segment timing specifies the time to pause between updates of the individual segments as they are plotted. This value can

be increased to slow down plot speed or to normalize plot speed because of PC performance differences. If you specify a value for the **occurrence refnum for abort** parameter, this **segment timing** value should be sufficiently long to allow the occurrence to be detected. The default is 100 ms.

- occurrence refnum for abort specifies whether the VI monitors the occurrence while updating the segments of a plot. If a valid occurrence is wired as an input, this VI monitors the occurrence and exit without completing any remaining plots if the occurrence is fired during the update. You can use this parameter to make the 3D graph utility responsive to user input. When this parameter is unwired, the VI completes a requested graph update without interruption. The default is not a valid occurrence.
- **clear graph** specifies whether to clear the 3D plot before plotting new data. Set this parameter to TRUE to clear the plot. The default is FALSE.
- **3D curve** specifies a reference to the ActiveX 3D graph that displays the plot. This connection is prewired when you select the VI from the front panel **controls** palette.
- **utoscale** specifies whether to apply autoscaling to the graph referenced in the **eye diagram reference** control. When this control is set to TRUE, the X axis is scaled to display exactly (**eye length** × *symbol period*) seconds, while the Y and Z axes are scaled equally and symmetrically about zero for the largest magnitude of the signal.

The default is FALSE (no autoscaling).

- **display size** specifies the number of segments plotted simultaneously on the 3D graph. If this value is less than the number of available segments in the waveform, the VI plots the additional segments by adding them one at a time to the existing plot. As new segments are added the oldest segments are removed. The default is 100.
 - Note Because plotting on a 3D graph can be computationally intensive, you can set this parameter to a value that is less than the total number of segments that are

available in the input waveform. Doing so limits the maximum number of segments displayed at one time.

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or

Explain Warning from the shortcut menu for more information about the error.
Details

Tip For a "clean" eye diagram with distinct, nonshifting transitions, the input waveform should have an integer-valued symbol-rate-to-sample-rate ratio.

This VI works in conjunction with a standard LabVIEW graph indicator and is available both from the front panel **Controls** palette (**Controls»Modulation»Digital**) and from the block diagram **Functions** palette (**Functions»Modulation»Digital»Utilities**).

When selected from the front panel **Controls** palette, this VI is automatically prewired to the appropriate graph. When this VI is placed from the **Functions** palette onto the block diagram, no connections are automatically wired and you must manually connect the graph indicator on the block diagram.

MT Format Constellation VI

Prepares a signal for presentation on a graph showing the detected symbol locations and the transitions between those symbols. The VI specifies a complex-valued waveform and displays a constellation plot of the waveform contents.

<u>Details</u>



- waveform specifies the recovered modulated data for plotting as a constellation.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (1) and quadrature-phase (Q) data, respectively.
- **samples per symbol** specifies the number of samples per symbol in the recovered waveform. The default is 0.
- **utoscale** specifies whether to apply autoscaling to the graph referenced in the **constellation reference** control. When this control is set to TRUE, the vertical and horizontal axes of the plot are scaled symmetrically about zero, with enough room to accommodate the largest signal value. The default is FALSE (not autoscaling).
- **constellation reference** specifies a reference to the graph wired to the **constellation graph** parameter. This connection is prewired when you drop the VI from the front panel **Controls** palette.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other

VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **constellation graph** returns a two-element array of XY data. The first element is the constellation; the second element is the oversampled data. Wire this parameter directly to an XY graph.
 - Note If you originally placed this VI onto the front panel from the **Controls** palette as a new waveform graph, the **constellation graph** parameter is prewired to the graph.
 - **constellation** returns an array of doubles representing the constellation plot.
 - **oversampled data** returns an array of doubles representing the oversampled data.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

This VI assumes that the input waveform is a digitally <u>modulated</u> complex <u>baseband</u> signal containing samples that fall on symbol boundaries with a sample rate that is an integer multiple of the **samples per symbol** value.

Although this VI performs no <u>demodulation</u> or frequency correction, you can use this VI to examine the output of many of the digital demodulation VIs. In cases where the demodulator can recover symbol timing, the **output complex waveform** parameter of the digital demodulation VIs contains a frequency-corrected signal appropriate for plotting with this VI.

This VI works in conjunction with a standard LabVIEW graph indicator and is available both from the front panel **Controls** palette (**Controls**»**Modulation**»**Digital**) and from the block diagram **Functions** palette (**Functions**»**Modulation**»**Digital**»**Utilities**).

When selected from the front panel **Controls** palette, this VI is automatically prewired to the appropriate graph. When this VI is placed from the **Functions** palette, no connections are automatically wired and you must manually connect the graph indicator on the block diagram.

MT Format Eye Diagram VI

Specifies a complex- or real-valued waveform, divides it into segments, and displays those segments as plots on a waveform graph. This VI determines the segment length based on the **symbol rate** and **eye length** parameters.

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Format Eye Diagram (complex)



- waveform specifies the recovered modulated data for plotting as an eye diagram.
 - **t0** specifies the trigger (start) time of the acquired **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.
- eye length specifies the number of symbol periods on the horizontal scale of the eye diagram. The default is 1.
- I/Q specifies whether to generate I data or Q data to the eye diagram.

I (0) Generates in-phase (I) data. This is the default value.Q (1) Generates quadrature (Q) data.

- eye diagram reference is reference to the graph wired to the eye diagram parameter. This connection is prewired when you place the VI from the front panel **Controls** palette.
- **utoscale** specifies whether to apply autoscaling to the graph referenced in the **eye diagram reference** control. When this control is set to TRUE, the vertical axis of the eye diagram is scaled to fit the graph, while the horizontal (time) axis is scaled to display exactly (**eye length** × *symbol period*) seconds. The default is FALSE (no autoscaling).

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- eye diagram returns an array of waveform segments for eye diagram display. Each segment has a time length of (eye length × *symbol period*) seconds. Wire this parameter directly to a waveform graph.
 - Note If you originally placed this VI onto the front panel from the **Controls** palette as a new waveform graph, the **eye diagram** parameter is prewired to the new waveform graph.
 - **t0** returns the trigger (start) time of the Y array.
 - **DEL** dt returns the time between values in the Y array.
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu

for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Format Eye Diagram (WDT)



- waveform specifies the modulated data for plotting as an eye diagram.
- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.
- **eye length** specifies the number of symbol periods on the horizontal scale of the eye diagram. The default is 1.
- eye diagram reference specifies a reference to the graph wired to the eye diagram parameter. This parameter aids graph autoscaling. This connection is prewired when you place the VI from the front panel **Controls** palette.
- **TTO autoscale** specifies whether to apply autoscaling to the graph referenced in the **eye diagram reference** control. When this control is set to TRUE, the vertical axis of the eye diagram is scaled to fit the graph, while the horizontal (time) axis is scaled to display exactly (**eye length** × *symbol period*) seconds. The default is FALSE (no autoscaling).
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about

the error.

- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- eye diagram returns an array of waveform segments for eye diagram display. Each segment has a time length of (eye length × *symbol period*) seconds. Wire this parameter directly to a waveform graph.
 - Note If you originally placed this VI onto the front panel from the **Controls** palette as a new waveform graph, the **eye diagram** parameter is prewired to the new waveform graph.
 - **t0** returns the trigger (start) time of the Y array.
 - **besilved the time between values in the Y array.**
 - Y returns the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Tip For a "clean" eye diagram with distinct, non-shifting transitions, the input waveform should have an integer-valued symbol-rate-to-sample-rate ratio.

The complex instance of this VI displays either the I or Q component of the input signal, depending on the selection you make on the I/Q parameter. The I and Q components can also be examined simultaneously over time on a 3D graph using the <u>MT Display 3DEye</u> VI.

This VI works in conjunction with a standard LabVIEW graph indicator and is available both from the front panel **Controls** palette (**Controls»Modulation»Digital**) and from the block diagram **Functions** palette (**Functions»Modulation»Digital»Utilities**).

When selected from the **Controls** palette to the front panel, this VI is automatically prewired to the appropriate graph. When this VI is placed from the **Functions** palette onto the block diagram, no connections are automatically wired and you must manually connect the graph indicator on the block diagram.

MT Format Trellis Diagram VI

Receives a complex-valued waveform, divides it into segments of (**trellis length** / **symbol rate**), and displays those segments as a **trellis diagram** on a standard LabVIEW waveform graph referenced by the **trellis reference** parameter.

Details



- waveform specifies the modulated data for plotting on a trellis diagram.
 - **t0** specifies the trigger (start) time of the **Y** array. The default is 0.0.
 - **dt** specifies the time interval between data points in the **Y** array. The default is 1.0.
 - Y specifies the complex-valued signal-only baseband modulated waveform. The real and imaginary parts of this complex data array correspond to the in-phase (I) and quadrature-phase (Q) data, respectively.
- **symbol rate** specifies the desired symbol rate, in hertz. The default is 1.0.

trellis length specifies the number of symbol periods on the horizontal scale of the trellis diagram. The default is 2.

- **modulation index** specifies the <u>modulation index</u> of the input waveform. The default is appropriate for MSK modulation. The default is 0.5.
- **trellis reference** specifies a reference to the graph wired to the **trellis diagram** parameter. This connection is prewired when you select the VI from the front panel **Controls** palette.
- **autoscale** specifies whether to apply autoscaling to the graph referenced in the **trellis reference** control. When this control is set to TRUE, the vertical axis is scaled symmetrically about zero, with

enough room to accommodate the largest deviation from zero, while the horizontal axis is scaled to display exactly (**trellis length** × *symbol period*) seconds. The default is FALSE (no autoscaling).

- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **trellis diagram** returns an array of waveform segments for trellis diagram display. Each segment has a time length of (**eye length** × *symbol period*) seconds. Wire this parameter directly to a waveform graph.
 - Note If you originally placed this VI onto the front panel from the **Controls** palette as a new waveform graph, the **trellis diagram** parameter is prewired to the diagram.
 - **ID** returns the time of first value in **phase** array.
 - **besilon dt** returns the time between values in the **phase** array.
 - **phase** returns the unwrapped **phase** of the input **waveform**.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu

for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Tip For a "clean" diagram with distinct, nonshifting transitions, the input waveform should have an integer-valued symbol-rate-to-sample-rate ratio.

This VI works in conjunction with a standard LabVIEW graph indicator and is available both from the front panel **Controls** palette (**Controls**»Modulation»Digital) and from the block diagram Functions palette (Functions»Modulation»Digital»Utilities).

When selected from the **Controls** palette to the front panel, this VI is automatically prewired to the appropriate graph. When this VI is placed from the **Functions** palette to the block diagram, no connections are automatically wired and you must manually connect the graph indicator on the block diagram.

MT Block Deinterleave (Matrix) VI

Performs deinterleaving by writing the input data into an m(rows) × n(columns) matrix, column by column, and then reading the data out row by row when the matrix is full. Typically, block interleaver and deinterleaver functions are used together.

<u>Details</u>



- **basis** data in specifies input data to be deinterleaved. Data is processed in blocks of *mn*, where *m* is the number of rows and *n* is the number of columns. If the number of data samples is less than block size *mn*, the block deinterleaver does not return any data if the VI is run once. If the data samples are continuously fed in and if the number of data samples is greater than but not a multiple of *mn*, the block deinterleaver buffers the remaining samples for use in the next iteration. For example, if *m*=3, *n*=4, and if the data samples are coming in number of 27, the block deinterleaver deinterleaves two blocks ($2 \times 12 = 24$) of data, and buffers 3 samples (27 - 24 = 3) for use in the next iteration. Data is read in by column and read out by row. The default is empty.
- **number of rows** specifies the number of rows of the block deinterleaver. The default is 1.
- **number of columns** specifies the number of columns of the block deinterleaver. The default is 1.
- **reset?** specifies how the VI handles buffered data. When **reset?** is set to TRUE, buffered data is cleared. Input parameters are checked on a first call to this VI. Any change in the input parameter values during subsequent iterations of the VI is reflected only if **reset?** is set to TRUE. In continuous operations, when the buffered data is appended in the beginning of data from next iteration, set **reset?** to FALSE. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other

VIs. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **data out** returns the output of the block deinterleave operation. The data generated of a block deinterleaver is the same as the data specified to the <u>block interleaver</u>.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

Refer to the following theoretical example for an illustration of the deinterleave operation.

If the **data in** parameter is entered as 0,5,10,1,6,11,2,7,12,3,8,13,4,9,14, the block could be visualized as follows:

0	1	2	3	4
5	6	7	8	0
10	11	12	13	14

The returned deinterleaved data in the **data out** parameter would be 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,...

MT Generate Fading Profile VI

Generates a user-specified Rayleigh or Rician fading profile for the complex baseband waveform. Call the <u>MT Apply Fading Profile</u> VI to apply the generated profile to a baseband I/Q waveform. You can use this VI for modeling <u>Rayleigh</u> and <u>Rician</u> fading profile distributions using Gans and Jakes models for characterizing flat-fading channels and time-varying frequency selective-fading channels.

•

Use the pull-down menu to select an instance of this VI.

Select an instance

Details

MT Rayleigh Fading profile (Jakes)

This polymorphic instance generates a Rayleigh flat-fading profile with an envelope that statistically obeys the Rayleigh distribution, using the Jakes fading model.



- **profile length** specifies the number of complex-valued fading profile samples (having Rayleigh-distributed envelopes) to generate. The default is 1,000.
- **sampling frequency** specifies the system sampling rate, in hertz. This rate is the product of the symbol rate × samples per symbol. The default is 1.0.
- **doppler spread** specifies the desired input Doppler spread f_m of the channel, in hertz. The default is 100.00 mHz.

This parameter denotes the measure of the spectral broadening caused by the time rate of change of the channel. Doppler spread is defined as the range of frequencies over which the received Doppler spectrum is essentially nonzero. When a pure sine tone at frequency f_c is transmitted, the received signal spectrum, called the Doppler spectrum, has components in the range $(f_c - f_m)$ to $(f_c + f_m)$. The Doppler spread is related to the mobile velocity *v*, carrier frequency f_c , and the speed of light *c* by the relation $f_m = v f_c/c$.

- Note The inverse of the Doppler spread $T_c = 1 / 16_{\pi} f_m$ (known as the coherence time) is the time duration over which the channel impulse response is essentially invariant.
- seed in specifies the initial state for generating the fading profile. If seed in is set to -1, the generated fading profile is randomly chosen during every call when reset? is set to TRUE. Otherwise, the generated fading profile returns the same set of fading coefficients when reset? is set to TRUE. The seed in value is used only for the first call or when reset? is set to TRUE. The default is -

1.

- **fading variance** specifies the desired variance of the complex-valued Rayleigh distributed fading profile. The default is 1.0.
- **reset?** specifies whether the fading profile generation is reset on subsequent calls to this VI. When **reset?** is set to TRUE, fading profile generation is reset on every call to this VI. Set **reset?** to FALSE to continue generating the fading profile from the previous iteration on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **fading profile** returns complex-valued coefficients of the generated fading profile size that equals **profile length**. Wire this parameter to the <u>MT Apply Fading Profile</u> VI to apply this fading profile to a baseband I/Q signal.
- **seed out** returns the internal state of the VI at the end of generation of the fading profile for the current iteration.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and

select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Rayleigh Fading profile (Gans)

This polymorphic instance generates a Rayleigh flat-fading profile with an envelope that statistically obeys the Rayleigh distribution, using the Gans fading model.



- **profile length** specifies the number of complex-valued fading profile samples (having Rayleigh-distributed envelopes) to generate. The default is 1,000.
- **sampling frequency** specifies the system sampling rate, in hertz. This rate is the product of the symbol rate × samples per symbol. The default is 1.0.
- **Description** doppler spread specifies the desired input Doppler spread f_m of the channel, in hertz. The default is 100.00 mHz.

This parameter denotes the measure of the spectral broadening caused by the time rate of change of the channel. Doppler spread is defined as the range of frequencies over which the received Doppler spectrum is essentially nonzero. When a pure sine tone at frequency f_c is transmitted, the received signal spectrum, called the Doppler spectrum, has components in the range $(f_c - f_m)$ to $(f_c + f_m)$. The Doppler spread is related to the mobile velocity *v*, carrier frequency f_c , and the speed of light *c* by the relation $f_m = v f_c/c$.

- Note The inverse of the Doppler spread $T_c = 1 / 16_{\pi} f_m$ (known as the coherence time) is the time duration over which the channel impulse response is essentially invariant.
- seed in specifies the initial state for generating the fading profile. If seed in is set to -1, the generated fading profile is randomly chosen during every call when reset? is set to TRUE. Otherwise, the generated fading profile returns the same set of fading coefficients when reset? is set to TRUE. The default is -1.

DBL

fading variance specifies the desired variance of the complexvalued Rayleigh distributed fading profile. The default is 1.0.

- **reset?** specifies whether the fading profile generation is reset on subsequent calls to this VI. When **reset?** is set to TRUE, fading profile generation is reset on every call to this VI. Set **reset?** to FALSE to continue generating the fading profile from the previous iteration on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **fading profile** returns complex-valued coefficients of the generated fading profile size that equals **profile length**. Wire this parameter to the <u>MT Apply Fading Profile</u> VI to apply this fading profile to a baseband I/Q signal.
- **seed out** returns the internal state of the VI at the end of generation of the fading profile for the current iteration. When **reset?** is set to FALSE, this state is used to continue the fading profile generation at the beginning of the next iteration.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and

select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

- status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Rician Fading profile (Jakes)

This polymorphic instance generates a Rician flat-fading profile whose envelope statistically obeys the Rician distribution, using the Jakes fading model.



- **profile length** specifies the number of complex-valued fading profile samples (having Rician-distributed envelopes) to generate. The default is 1,000.
- **sampling frequency** specifies the system sampling rate, in hertz. This rate is the product of the symbol rate × samples per symbol. The default is 1.0.
- **doppler spread** specifies the desired input Doppler spread f_m of the channel, in hertz. The default is 100.00 mHz.

This parameter denotes the measure of the spectral broadening caused by the time rate of change of the channel. Doppler spread is defined as the range of frequencies over which the received Doppler spectrum is essentially nonzero. When a pure sine tone at frequency f_c is transmitted, the received signal spectrum, called the Doppler spectrum, has components in the range $(f_c - f_m)$ to $(f_c + f_m)$. The Doppler spread is related to the mobile velocity v, carrier frequency f_c , and the speed of light c by the relation $f_m = v f_c/c$.

 $\overline{\mathbb{N}}$

Note The inverse of the Doppler spread $T_c = 1 / 16\pi f_m$ (known as the coherence time) is the time duration over which the channel impulse response is essentially invariant.

seed in specifies the initial state for generating the fading profile. If seed in is set to -1, the generated fading profile is randomly chosen during every call when reset? is set to TRUE. Otherwise, the generated fading profile returns the same set of fading coefficients when reset? is set to TRUE. The seed in value is used

only for the first call or when **reset?** is set to TRUE. The default is – 1.

rician parameter K specifies the desired ratio, in dB, of the dominant line-of-sight (LOS) path to the scattering component. A large positive value of *K* represents a strongly additive white Gaussian noise channel, while a large negative value of *K* represents a Rayleigh fading (predominantly scattering) channel. Mathematically, the Rician parameter *K* is defined as

K (dB) = 10 × log{Power[LOS component]/Power[scattering component]}

The Rician fading profile is generated by adding a DC specular component to a Rayleigh distributed scattering component. *var* denotes the requested **fading variance** — the variance of the underlying Rayleigh fading profile. The amplitude of the specular DC component equals

 $A = \sqrt{var \times K}$

By varying K, you can control the extent of the scattering component relative to the LOS component of fading. For a strongly Gaussian channel, K approaches infinity, while K < 0 indicates a strongly scattering (Rayleigh fading) channel. The default is 0 dB.

- **fading variance** specifies the desired variance of the complexvalued Rician distributed fading profile. The default is 1.0.
- reset? specifies whether the fading profile generation is reset on subsequent calls to this VI. When reset? is set to TRUE, fading profile generation is reset on every call to this VI. Set reset? to FALSE to continue generating the fading profile from the previous iteration on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain**

Error or **Explain Warning** from the shortcut menu for more information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **fading profile** returns complex-valued coefficients of the generated fading profile size that equals **profile length**. Wire this parameter to the <u>MT Apply Fading Profile</u> VI to apply this fading profile to a baseband I/Q signal.
- **seed out** returns the internal state of the VI at the end of generation of the fading profile for the current iteration.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Rician Fading profile (Gans)

This polymorphic instance generates a Rician flat-fading profile for the multipath channel. The envelope for the first path statistically obeys the Rician distribution, while the envelope for the remaining path obeys the Rayleigh distribution implemented using the Gans model.



- **profile length** specifies the number of complex-valued fading profile samples (having Rician-distributed envelopes) to generate. The default is 1,000.
- **sampling frequency** specifies the system sampling rate, in hertz. This rate is the product of the symbol rate × samples per symbol. The default is 1.0.
- **doppler spread** specifies the desired input Doppler spread f_m of the channel, in hertz. The default is 100.00 mHz.

This parameter denotes the measure of the spectral broadening caused by the time rate of change of the channel. Doppler spread is defined as the range of frequencies over which the received Doppler spectrum is essentially nonzero. When a pure sine tone at frequency f_c is transmitted, the received signal spectrum, called the Doppler spectrum, has components in the range $(f_c - f_m)$ to $(f_c + f_m)$. The Doppler spread is related to the mobile velocity v, carrier frequency f_c , and the speed of light c by the relation $f_m = v f_c/c$.

Note The inverse of the Doppler spread $T_c = 1 / 16_{\pi} f_m$ (known as the coherence time) is the time duration over which the channel impulse response is essentially invariant.

seed in specifies the initial state for generating the fading profile. If seed in is set to -1, the generated fading profile is randomly chosen during every call when reset? is set to TRUE. Otherwise, the generated fading profile returns the same set of fading coefficients when **reset?** is set to TRUE. The default is -1.

rician parameter K specifies the desired ratio, in dB, of the dominant line-of-sight (LOS) path to the scattering component. A large positive value of *K* represents a strongly additive white Gaussian noise channel, while a large negative value of *K* represents a Rayleigh fading (predominantly scattering) channel. Mathematically, the Rician parameter *K* is defined as

K (dB) = 10 × log{Power[LOS component]/Power[scattering component]}

The Rician fading profile is generated by adding a DC specular component to a Rayleigh distributed scattering component. *var* denotes the requested **fading variance** — the variance of the underlying Rayleigh fading profile. The amplitude of the specular DC component equals

 $A = \sqrt{var \times K}$

By varying K, you can control the extent of the scattering component relative to the LOS component of fading. For a strongly Gaussian channel, K approaches infinity, while K < 0 indicates a strongly scattering (Rayleigh fading) channel. The default is 0 dB.

- **fading variance** specifies the desired variance of the complexvalued Rician distributed fading profile. The default is 1.0.
- **reset?** specifies whether the fading profile generation is reset on subsequent calls to this VI. When **reset?** is set to TRUE, fading profile generation is reset on every call to this VI. Set **reset?** to FALSE to continue generating the fading profile from the previous iteration on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more

information about the error.

- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **fading profile** returns complex-valued coefficients of the generated fading profile size that equals **profile length**. Wire this parameter to the <u>MT Apply Fading Profile</u> VI to apply this fading profile to a baseband I/Q signal.
- **seed out** returns the internal state of the VI at the end of generation of the fading profile for the current iteration. When **reset?** is set to FALSE, this state is used to continue the fading profile generation at the beginning of the next iteration.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** is the error or warning code. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Rician Selective Fading profile (Gans)

This polymorphic instance generates a Rician selective fading profile for the multipath channel. The envelope for the first path statistically obeys the Rician distribution, while the envelope for the remaining path obeys the Raleigh distribution implemented using the Gans model.



- **profile length** specifies the number of complex-valued fading profile samples (having Rician-distributed envelopes) to generate. The default is 1,000.
- **sampling frequency** specifies the system sampling rate, in hertz. This rate is the product of the *symbol rate* × *samples per symbol*. The default is 1.0.
- **doppler spread** specifies the desired input Doppler spread f_m of the channel, in hertz. The default is 100.00 mHz.

This parameter denotes the measure of the spectral broadening caused by the time rate of change of the channel. Doppler spread is defined as the range of frequencies over which the received Doppler spectrum is essentially nonzero. When a pure sine tone at frequency f_c is transmitted, the received signal spectrum, called the Doppler spectrum, has components in the range $(f_c - f_m)$ to $(f_c + f_m)$. The Doppler spread is related to the mobile velocity *v*, carrier frequency f_c , and the speed of light *c* by the relation $f_m = v f_c/c$.

 $\overline{\mathbb{N}}$

Note The inverse of the Doppler spread $T_c = 1 / 16\pi f_m$ (known as the coherence time) is the time duration over which the channel impulse response is essentially invariant.

seed in specifies the initial state for generating the fading profile. If seed in is set to -1, the generated fading profile is randomly chosen during every call when reset? is set to TRUE. Otherwise, the generated fading profile returns the same set of fading coefficients when **reset?** is set to TRUE. The default is -1.

- **number of paths** specifies the number of paths in the simulated multipath channel. A fading profile is generated for each of these paths.
- **rician parameter K** specifies the desired ratio, in dB, of the dominant line-of-sight (LOS) path to the scattering component. A large positive value of *K* represents a strongly additive white Gaussian noise channel, while a large negative value of *K* represents a Rayleigh fading (predominantly scattering) channel. Mathematically, the Rician parameter *K* is defined as

K (dB) = 10 × log{Power[LOS component]/Power[scattering component]}

The Rician fading profile is generated by adding a DC specular component to a Rayleigh distributed scattering component. *var* denotes the requested **fading variance** — the variance of the underlying Rayleigh fading profile. The amplitude of the specular DC component equals

 $A = \sqrt{var \times K}$

By varying K, you can control the extent of the scattering component relative to the LOS component of fading. For a strongly Gaussian channel, K approaches infinity, while K < 0 indicates a strongly scattering (Rayleigh fading) channel. The default is 0 dB.

- **fading variance** specifies the desired variance of the complexvalued Rician distributed fading profile. The default is 1.0.
- **reset?** specifies whether the fading profile generation is reset on subsequent calls to this VI. When **reset?** is set to TRUE, fading profile generation is reset on every call to this VI. Set **reset?** to FALSE to continue generating the fading profile from the previous iteration on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **fading profile** returns a two-dimensional array, for each path, of complex-valued coefficients of the generated fading profile. The number of rows corresponds to the number of paths in the channel, and the number of columns is equal to the profile length. Wire this parameter to the <u>MT Apply Fading Profile</u> VI to apply this fading profile to a baseband I/Q signal.
- **seed out** returns the internal state of the VI at the end of generation of the fading profile for the current iteration. When **reset?** is set to FALSE, this state is used to continue the fading profile generation at the beginning of the next iteration.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

Source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
MT Rician Selective Fading profile (Jakes)

This polymorphic instance generates a Rician selective-fading profile for the multipath channel. The envelope for the first path statistically obeys the Rician distribution, while the envelope for the remaining path obeys the Rayleigh distribution implemented using the Jakes model.



- **profile length** specifies the number of complex-valued fading profile samples (having Rician-distributed envelopes) to generate. The default is 1,000.
- **sampling frequency** specifies the system sampling rate, in hertz. This rate is the product of the *symbol rate* × *samples per symbol*. The default is 1.0.
- **doppler spread** specifies the desired input Doppler spread f_m of the channel, in hertz. The default is 100.00 mHz.

This parameter denotes the measure of the spectral broadening caused by the time rate of change of the channel. Doppler spread is defined as the range of frequencies over which the received Doppler spectrum is essentially nonzero. When a pure sine tone at frequency f_c is transmitted, the received signal spectrum, called the Doppler spectrum, has components in the range $(f_c - f_m)$ to $(f_c + f_m)$. The Doppler spread is related to the mobile velocity *v*, carrier frequency f_c , and the speed of light *c* by the relation $f_m = v f_c/c$.

 \mathbb{N}

Note The inverse of the Doppler spread $T_c = 1 / 16_{\pi} f_m$ (known as the coherence time) is the time duration over which the channel impulse response is essentially invariant.

seed in specifies the initial state for generating the fading profile. If seed in is set to -1, the generated fading profile is randomly chosen during every call when reset? is set to TRUE. Otherwise, the generated fading profile returns the same set of fading coefficients when **reset?** is set to TRUE. The **seed in** value is used only for the first call or when **reset?** is set to TRUE. The default is -1.

- **number of paths** specifies the number of paths in the simulated multipath channel. A fading profile is generated for each of these paths.
- **rician parameter K** specifies the desired ratio, in dB, of the dominant line-of-sight (LOS) path to the scattering component. A large positive value of *K* represents a strongly additive white Gaussian noise channel, while a large negative value of *K* represents a Rayleigh fading (predominantly scattering) channel. Mathematically, the Rician parameter *K* is defined as

K (dB) = 10 × log{Power[LOS component]/Power[scattering component]}

The Rician fading profile is generated by adding a DC specular component to a Rayleigh distributed scattering component. *var* denotes the requested **fading variance** — the variance of the underlying Rayleigh fading profile. The amplitude of the specular DC component equals

 $A = \sqrt{var \times K}$

By varying K, you can control the extent of the scattering component relative to the LOS component of fading. For a strongly Gaussian channel, K approaches infinity, while K < 0 indicates a strongly scattering (Rayleigh fading) channel. The default is 0 dB.

- **fading variance** specifies the desired variance of the complexvalued Rician distributed fading profile. The default is 1.0.
- **reset?** specifies whether the fading profile generation is reset on subsequent calls to this VI. When **reset?** is set to TRUE, fading profile generation is reset on every call to this VI. Set **reset?** to FALSE to continue generating the fading profile from the previous iteration on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain

Error or **Explain Warning** from the shortcut menu for more information about the error.

- **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **fading profile** returns a two-dimensional array, for each path, of complex-valued coefficients of the generated fading profile. The number of rows corresponds to the number of paths in the channel, and the number of columns is equal to the profile length. Wire this parameter to the <u>MT Apply Fading Profile</u> VI to apply this fading profile to a baseband I/Q signal.
- **seed out** returns the internal state of the VI at the end of generation of the fading profile for the current iteration.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.

source describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Rayleigh Selective Fading profile (Gans)

This polymorphic instance generates a selective-fading profile for the multipath channel. The envelope for each path statistically obeys the Rayleigh distribution implemented using the Gans fading model.



- **profile length** specifies the number of complex-valued fading profile samples (having Rayleigh-distributed envelopes) to generate. The default is 1,000.
- **sampling frequency** specifies the system sampling rate, in hertz. This rate is the product of the symbol rate × samples per symbol. The default is 1.0.
- **doppler spread** specifies the desired input Doppler spread f_m of the channel, in hertz. The default is 100.00 mHz.

This parameter denotes the measure of the spectral broadening caused by the time rate of change of the channel. Doppler spread is defined as the range of frequencies over which the received Doppler spectrum is essentially nonzero. When a pure sine tone at frequency f_c is transmitted, the received signal spectrum, called the Doppler spectrum, has components in the range $(f_c - f_m)$ to $(f_c + f_m)$. The Doppler spread is related to the mobile velocity v, carrier frequency f_c , and the speed of light c by the relation $f_m = v f_c/c$.

 $\overline{\mathbb{N}}$

Note The inverse of the Doppler spread $T_c = 1 / 16_{\pi} f_m$ (known as the coherence time) is the time duration over which the channel impulse response is essentially invariant.

seed in specifies the initial state for generating the fading profile. If seed in is set to -1, the generated fading profile is randomly chosen during every call when reset? is set to TRUE. Otherwise, the generated fading profile returns the same set of fading coefficients when reset? is set to TRUE. The default is -1.

- **number of paths** specifies the number of paths in the simulated multipath channel. A fading profile is generated for each of these paths.
- **fading variance** specifies the desired variance of the complexvalued Rayleigh distributed fading profile. The default is 1.0.
- **reset?** specifies whether the fading profile generation is reset on subsequent calls to this VI. When **reset?** is set to TRUE, fading profile generation is reset on every call to this VI. Set **reset?** to FALSE to continue generating the fading profile from the previous iteration on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
- **fading profile** returns a two-dimensional array, for each path, of complex-valued coefficients of the generated fading profile. The number of rows corresponds to the number of paths in the channel, and the number of columns is equal to the profile length. Wire this parameter to the <u>MT Apply Fading Profile</u> VI to apply this fading profile to a baseband I/Q signal.
- **seed out** returns the internal state of the VI at the end of generation

of the fading profile for the current iteration. When **reset?** is set to FALSE, this state is used to continue the fading profile generation at the beginning of the next iteration.

- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **Source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

MT Rayleigh Selective Fading profile (Jakes)

This polymorphic instance generates a Rayleigh selective-fading profile for the multipath channel. The envelope for each path statistically obeys the Rayleigh distribution implemented using the Jakes fading model.



- **profile length** specifies the number of complex-valued fading profile samples (having Rayleigh-distributed envelopes) to generate. The default is 1,000.
- **sampling frequency** specifies the system sampling rate, in hertz. This rate is the product of the symbol rate × samples per symbol. The default is 1.0.
- **doppler spread** specifies the desired input Doppler spread f_m of the channel, in hertz. The default is 100.00 mHz.

This parameter denotes the measure of the spectral broadening caused by the time rate of change of the channel. Doppler spread is defined as the range of frequencies over which the received Doppler spectrum is essentially nonzero. When a pure sine tone at frequency f_c is transmitted, the received signal spectrum, called the Doppler spectrum, has components in the range $(f_c - f_m)$ to $(f_c + f_m)$. The Doppler spread is related to the mobile velocity v, carrier frequency f_c , and the speed of light c by the relation $f_m = v f_c/c$.

 $\overline{\mathbb{N}}$

Note The inverse of the Doppler spread $T_c = 1 / 16_{\pi} f_m$ (known as the coherence time) is the time duration over which the channel impulse response is essentially invariant.

seed in specifies the initial state for generating the fading profile. If seed in is set to -1, the generated fading profile is randomly chosen during every call when reset? is set to TRUE. Otherwise, the generated fading profile returns the same set of fading coefficients when reset? is set to TRUE. The seed in value is used

only for the first call or when **reset?** is set to TRUE. The default is -1.

- **number of paths** specifies the number of paths in the simulated multipath channel. A fading profile is generated for each of these paths.
- **fading variance** specifies the desired variance of the complexvalued Rayleigh distributed fading profile. The default is 1.0.
- **reset?** specifies whether the fading profile generation is reset on subsequent calls to this VI. When **reset?** is set to TRUE, fading profile generation is reset on every call to this VI. Set **reset?** to FALSE to continue generating the fading profile from the previous iteration on subsequent calls. The default is TRUE.
- error in (no error) can accept error information wired from previously called VIs. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs. Right-click the front panel error in control and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **code** identifies the error or warning code. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error in** control and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

fading profile returns a two-dimensional array, for each path, of complex-valued coefficients of the generated fading profile. The number of rows corresponds to the number of paths in the channel, and the number of columns is equal to the profile length. Wire this parameter to the <u>MT Apply Fading Profile</u> VI to apply this fading

profile to a baseband I/Q signal.

- **seed out** returns the internal state of the VI at the end of generation of the fading profile for the current iteration.
- error out passes error or warning information out of a VI to be used by other VIs. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - code is the error or warning code. Right-click the front panel error out indicator and select Explain Error or Explain Warning from the shortcut menu for more information about the error.
 - **source** describes the origin of the error or warning. Right-click the front panel **error out** indicator and select **Explain Error** or **Explain Warning** from the shortcut menu for more information about the error.

Details

The selective fading profile instances of the MT Generate Fading Profile VI generate Rayleigh or Rician distributed fading profiles for the specified **number of paths** based on the Jakes or Gans model. The generated **fading profile** is a two-dimensional array in which the number of rows is equal to the **number of paths**, and the number of columns is equal to the **profile length**. The generated profile is passed to the <u>MT Apply Fading Profile</u> VI.

Fading profiles can be characterized as multiplicative models that attenuate the magnitude and phase of the transmitted modulated I/Q baseband signal. The faded waveform can be used to test receiver immunity to fading channels. This VI models the following fading profile distributions:

• **Rayleigh**—Describes a flat-fading channel characterized by a single-tap impulse response with a time-varying envelope that is Rayleigh-distributed. This model describes the statistical time-varying nature of the received envelope of a flat fading channel or the envelope of an individual multipath component. The Rayleigh distribution has a probability density function (PDF) given by the following equation:

$$p(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) u(r)$$

where *r* is the specified **fading variance**.

• **Rician**—Describes a time-varying channel with an envelope that follows a Rician distribution. The channel can be characterized by a single-tap impulse response comprising a dominant line-of-sight (LOS) path superimposed on a random multipath. The limiting case of a Rician fading channel (when the LOS path is much weaker than the random multipath) is the Rayleigh fading channel. The Rician distribution is given by

$$p(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2 + A^2}{2\sigma^2}\right) I_0\left(\frac{Ar}{\sigma^2}\right) u(r)$$

where A denotes the peak amplitude of the dominant signal, I_0 () denotes the modified Bessel function of the first kind and zeroorder, and *r* is the specified **fading variance**. The VI implements the Rayleigh/Rician fading profiles using two methods, the Jakes model and the Gans model. Each model is implemented by separate instances of this polymorphic VI.

 Jakes—Deterministic method that simulates time-correlated Rayleigh fading waveforms. The model assumes that N equalstrength rays arrive at a moving receiver with uniformly distributed arrival angles, such that ray *n* experiences a Doppler shift defined by the following equation:

$$\omega_n = \omega_m \cos(\alpha_n)$$

where

 $\omega_m = 2\pi f_m$, and

 $\alpha_m = 2\pi \frac{(n-0.5)}{N}$ represents the arrival angle of the ray *n*.

• Gans—Generates the Rayleigh fading profile by passing quadrature Gaussian I/Q components through a Doppler filter with a U-shaped power spectral density profile. For the selective fading model, the implementation ensures that the generated fading profile for all paths is uncorrelated.