

# Wireless HDL Toolbox™ Release Notes



# MATLAB® & SIMULINK®



## How to Contact MathWorks



Latest news: [www.mathworks.com](http://www.mathworks.com)  
Sales and services: [www.mathworks.com/sales\\_and\\_services](http://www.mathworks.com/sales_and_services)  
User community: [www.mathworks.com/matlabcentral](http://www.mathworks.com/matlabcentral)  
Technical support: [www.mathworks.com/support/contact\\_us](http://www.mathworks.com/support/contact_us)



Phone: 508-647-7000



The MathWorks, Inc.  
1 Apple Hill Drive  
Natick, MA 01760-2098

### *Wireless HDL Toolbox™ Release Notes*

© COPYRIGHT 2017 - 2020 by The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

### **Trademarks**

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

### **Patents**

MathWorks products are protected by one or more U.S. patents. Please see [www.mathworks.com/patents](http://www.mathworks.com/patents) for more information.

## R2020a

<b>LTE HDL Toolbox name change to Wireless HDL Toolbox</b> .....	<b>1-2</b>
<b>5G NR Signal Synchronization Reference Application: Use primary and secondary synchronization signals (PSS and SSS) to detect connection to valid cell</b> .....	<b>1-2</b>
<b>5G NR Polar Decoder and Encoder Blocks: Implement polar error correction algorithm according to 5G New Radio (NR) standard</b> .....	<b>1-2</b>
<b>5G NR LDPC Decoder and Encoder Blocks: Implement low-density parity checking according to 5G New Radio (NR) standard</b> .....	<b>1-3</b>
<b>OFDM Modulator Block: Modulate orthogonal frequency division multiplexed symbols for custom communication protocols</b> .....	<b>1-3</b>
<b>OFDM Channel Estimator Block: Estimate OFDM channel using LS channel estimation algorithm</b> .....	<b>1-3</b>
<b>RS Decoder Block: Decode Reed-Solomon codeword to recover message data</b> .....	<b>1-3</b>
<b>High-throughput OFDM Demodulation: Increase throughput using frame-based input</b> .....	<b>1-3</b>
<b>Functionality being removed or changed</b> .....	<b>1-3</b>
ltehdlFramesToSamples function will be removed .....	<b>1-3</b>
ltehdlSamplesToFrames function will be removed .....	<b>1-3</b>

## R2019b

<b>MIB and SIB1 Decoder Enhancements: Increase robustness of receiver reference applications</b> .....	<b>2-2</b>
<b>OFDM Demodulator Block: Demodulate orthogonal frequency division multiplexed symbols for custom communication protocols</b> .....	<b>2-2</b>
<b>FFT 1536 Block: Optimize resource usage for implementing LTE signals with 15 MHz bandwidth option</b> .....	<b>2-2</b>

<b>LTE and 5G NR Symbol Demodulator Blocks: Demodulate complex PSK or QAM symbols for LTE or 5G NR</b> .....	2-2
<b>Convolutional Encoder and Puncturer Blocks: Encode bit streams with continuous, terminated, and truncated modes for custom communication protocols</b> .....	2-2
<b>LTE Multiantenna Transmitter Reference Application: Generate waveforms for transmission on 1, 4, or 8 antennas</b> .....	2-3
<b>Sample Rate Conversion for LTE Receiver: Convert data sample rate to LTE sample rate of 30.72 Msps</b> .....	2-3

## R2019a

<b>OFDM Modulator Block: Modulate orthogonal frequency division multiplexing symbols according to the LTE standard</b> .....	3-2
<b>LTE Transmitter Reference Application: Generate baseband waveform including PSS, SSS, cell-specific reference, and MIB</b> .....	3-2
<b>LTE and 5G NR Symbol Modulator Blocks: Modulate message bits according to the LTE or 5G standard</b> .....	3-2
<b>1536-Point FFT Example: Implement a 1536-point FFT using a single 512-point FFT block</b> .....	3-2
<b>Variable sample-rate input for OFDM Demodulator block</b> .....	3-2
<b>TDD Support for SIB Recovery Reference Application: Recover System Information Block type 1 (SIB1) data from time-division duplex LTE signals</b> .....	3-3
<b>MIB recovery with strongest cell detection</b> .....	3-3
<b>Merge of MIB and MIB TDD reference applications into single model</b> ..	3-3
<b>Separation of LTE PSS and SSS detection into LTE Cell Search reference application</b> .....	3-3
<b>Accelerate Measurement of Bit-Error-Rate (BER)</b> .....	3-3

## R2018b

<b>SIB1 Reference Application: Implement an LTE System Information Block type 1 (SIB1) recovery subsystem on your FPGA or ASIC</b> .....	4-2
--	-----

<b>Viterbi Decoder and Depuncturer Blocks: Decode convolutionally encoded bit streams using the Viterbi algorithm with puncturing, terminated, and truncated modes</b> .....	4-2
<b>Reset port for OFDM Demodulator block</b> .....	4-2
<b>Variable-size FFT example with arbitrary input pattern</b> .....	4-2

## R2018a

<b>TDD Support for MIB Recovery Reference Application: Recover master information block data from time-division duplex LTE signals</b> .....	5-2
<b>5G Filtered OFDM Modulation Example: Implement a filtered orthogonal frequency division multiplexing transmitter in hardware</b> .....	5-2
<b>Gold Sequence Generator Block: Generate LTE-specific Gold sequences for channel estimation and descrambling</b> .....	5-2
<b>OFDM Demodulator Block: Demodulate orthogonal frequency division multiplexing symbols according to the LTE standard</b> .....	5-2
<b>Variable-Size FFT Example: Implement flexible-bandwidth FFT using a single FFT block</b> .....	5-2

## R2017b

<b>Standard-compliant LTE Simulink blocks</b> .....	6-2
<b>Downlink detector and MIB recovery reference application</b> .....	6-2
<b>HDL code generation support</b> .....	6-2
<b>Frame-to-sample and sample-to-frame conversions</b> .....	6-2
<b>FPGA-in-the-loop verification on Xilinx or Intel boards</b> .....	6-2
<b>Algorithm prototyping on Xilinx Zynq-based hardware</b> .....	6-2



# R2020a

---

**Version: 2.0**

**New Features**

**Bug Fixes**

**Compatibility Considerations**

## LTE HDL Toolbox name change to Wireless HDL Toolbox

The Wireless HDL Toolbox™ name reflects the expansion of the toolbox to include algorithms for 5G New Radio (NR) and general communications designs along with LTE algorithms.

This table shows the updated library hierarchy.

Old Libraries	Updated Libraries
<ul style="list-style-type: none"> <li>▼ LTE HDL Toolbox               <ul style="list-style-type: none"> <li>Error Detection and Correction</li> <li>General Communications</li> <li>I/O Interfaces</li> <li>Modulation</li> <li>Utilities</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>▼ Wireless HDL Toolbox               <ul style="list-style-type: none"> <li>Error Detection and Correction</li> <li>I/O Interfaces</li> <li>Modulation</li> <li>Utilities</li> </ul> </li> </ul>

All blocks from the LTE HDL Toolbox™ libraries are available in the Wireless HDL Toolbox libraries. The blocks formerly in the **General Communications** library are now split between the **Error Detection and Correction** and **Modulation** libraries. This table shows the updated location of each of these blocks.

Block	Updated Library
Convolutional Encoder	Error Detection and Correction
Depuncturer	Error Detection and Correction
OFDM Demodulator	Modulation
Puncturer	Error Detection and Correction
Viterbi Decoder	Error Detection and Correction

## 5G NR Signal Synchronization Reference Application: Use primary and secondary synchronization signals (PSS and SSS) to detect connection to valid cell

The “NR HDL Cell Search” reference application performs primary and secondary signal synchronization (PSS and SSS) as per the 5G NR standard. This design supports HDL code generation with HDL Coder™ and is ready for deployment to hardware.

The “NR HDL Cell Search MATLAB Reference” example shows how to model the 5G NR cell search hardware algorithm in MATLAB as a step towards developing a Simulink HDL implementation. The NR HDL Cell Search example uses this MATLAB reference to verify the Simulink reference application model.

## 5G NR Polar Decoder and Encoder Blocks: Implement polar error correction algorithm according to 5G New Radio (NR) standard

The NR Polar Encoder and NR Polar Decoder blocks implement polar error correction codes according to the 5G NR standard. Both blocks support HDL code generation with HDL Coder.



---

## **5G NR LDPC Decoder and Encoder Blocks: Implement low-density parity checking according to 5G New Radio (NR) standard**

The NR LDPC Decoder block implements layered belief propagation with the min-sum approximation algorithm for decoding low-density parity-check (LDPC) codes. The NR LDPC Encoder block encodes LDPC codes as per the 5G NR standard. Both blocks support channel coding of downlink and uplink shared channels and paging channel as per the 5G NR standard. These blocks use hardware-friendly control signals and support HDL code generation with HDL Coder.

## **OFDM Modulator Block: Modulate orthogonal frequency division multiplexed symbols for custom communication protocols**

The OFDM Modulator block supports standard and custom communication protocols. The block supports HDL code generation with HDL Coder.

## **OFDM Channel Estimator Block: Estimate OFDM channel using LS channel estimation algorithm**

The OFDM Channel Estimator block estimates an OFDM channel by using the least square (LS) channel estimation algorithm. The block supports averaging and interpolation techniques to improve the accuracy of LS estimation. The block uses hardware-friendly control signals and supports HDL code generation with HDL Coder.

## **RS Decoder Block: Decode Reed-Solomon codeword to recover message data**

The RS Decoder block decodes and recovers a message vector from a Reed-Solomon (RS) codeword. The block supports continuous processing of frames and contains a control bus to handle this frame data. This block provides an interface and architecture for HDL code generation with HDL Coder.

## **High-throughput OFDM Demodulation: Increase throughput using frame-based input**

The OFDM Demodulator block now supports a frame-based input. The block uses hardware-friendly control signals and supports HDL code generation with HDL Coder.

## **Functionality being removed or changed**

### **ltehdlFramesToSamples function will be removed**

*Warns*

`ltehdlFramesToSamples` will be removed in a future release. Use `whdlFramesToSamples` instead. To update your code, replace the instances of `ltehdlFramesToSamples` with `whdlFramesToSamples`. The functionality of the two functions is the same.

### **ltehdlSamplesToFrames function will be removed**

*Warns*

`ltehdlSamplesToFrames` will be removed in a future release. Use `whdlSamplesToFrames` instead. To update your code, replace the instances of `ltehdlSamplesToFrames` with `whdlSamplesToFrames`. The functionality of the two functions is the same.

# R2019b

---

**Version: 1.4**

**New Features**

**Bug Fixes**

## **MIB and SIB1 Decoder Enhancements: Increase robustness of receiver reference applications**

The LTE HDL MIB Recovery and LTE HDL SIB1 Recovery reference applications now support 2x transmit diversity decoding. These enhancements increase the decoding success rate for low quality input signals.

## **OFDM Demodulator Block: Demodulate orthogonal frequency division multiplexed symbols for custom communication protocols**

The OFDM Demodulator block demodulates OFDM symbols for custom communication protocols. The block provides an interface and architecture for HDL code generation and hardware deployment.

## **FFT 1536 Block: Optimize resource usage for implementing LTE signals with 15 MHz bandwidth option**

The LTE standard provides multiple transmission bandwidth options ranging from 1.4 MHz to 20 MHz. For the 15 MHz bandwidth option, you can use a 1536-point fast Fourier transform (FFT) for OFDM modulation and demodulation. The FFT 1536 block implements a 1536-point FFT by using a single 512-point FFT. The block provides an interface and architecture for HDL code generation and hardware deployment.

## **LTE and 5G NR Symbol Demodulator Blocks: Demodulate complex PSK or QAM symbols for LTE or 5G NR**

The LTE Symbol Demodulator and NR Symbol Demodulator blocks demodulate complex phase-shift keying (PSK) or quadrature amplitude modulation (QAM) symbols by using either hard or soft decision decoding. They return message bits or log-likelihood ratios (LLR) based on the modulation type. Each block support these modulation types specified by LTE standard TS 36.211 and 3GPP 5G standard TS 38.211, respectively.

- LTE Symbol Demodulator: BPSK, QPSK, and 16, 64, and 256 QAM
- NR Symbol Demodulator:  $\pi/2$ -BPSK, BPSK, QPSK, and 16, 64, and 256 QAM

Each block provides an interface and architecture for HDL code generation and hardware deployment.

## **Convolutional Encoder and Puncturer Blocks: Encode bit streams with continuous, terminated, and truncated modes for custom communication protocols**

You can customize the parameters of the Convolutional Encoder and Puncturer blocks to support most communication protocols, including continuous, terminated, and truncated modes. These blocks use hardware-friendly control signals and support HDL code generation with HDL Coder.

---

## **LTE Multiantenna Transmitter Reference Application: Generate waveforms for transmission on 1, 4, or 8 antennas**

The LTE HDL PBCH Transmitter example is extended to support 1, 4, or 8 antennas.

This example generates the baseband waveform specified by LTE standard TS 36.211. The waveform includes the primary synchronization signal (PSS), secondary synchronization signal (SSS), cell-specific reference signals (Cell-RS), and the master information block (MIB) for transmission through the physical broadcast channel (PBCH). The design supports HDL code generation with HDL Coder and is ready for deployment to hardware.

## **Sample Rate Conversion for LTE Receiver: Convert data sample rate to LTE sample rate of 30.72 Msps**

The Sample Rate Conversion for an LTE Receiver example converts input waveforms to the 30.72 Msps sample rate required by the LTE receiver. The model is compatible with the other LTE receiver reference applications and supports HDL code generation with HDL Coder.



# R2019a

---

**Version: 1.3**

**New Features**

**Bug Fixes**

## **OFDM Modulator Block: Modulate orthogonal frequency division multiplexing symbols according to the LTE standard**

The OFDM Modulator block implements an algorithm for modulating LTE signals specified by LTE standard TS 36.212. It provides an interface and architecture for HDL code generation and hardware deployment. The block modulates an encoded resource grid into time-domain OFDM samples.

## **LTE Transmitter Reference Application: Generate baseband waveform including PSS, SSS, cell-specific reference, and MIB**

The LTE HDL PBCH Transmitter example generates the baseband waveform specified by LTE standard TS 36.211. The waveform includes the primary synchronization signal (PSS), secondary synchronization signal (SSS), cell-specific reference signals (Cell-RS), and the master information block (MIB) for transmission through the Physical Broadcast Channel (PBCH). The design supports HDL code generation with HDL Coder and is ready for deployment to hardware.

## **LTE and 5G NR Symbol Modulator Blocks: Modulate message bits according to the LTE or 5G standard**

The LTE Symbol Modulator block and the NR Symbol Modulator block map groups of bits to complex data symbols according to a dynamic modulation scheme. These supported modulation schemes are specified by LTE standard TS 36.211 and 3GPP 5G standard TS 38.211.

- LTE Symbol Modulator: BPSK, QPSK, 16/64/256-QAM
- NR Symbol Modulator:  $\pi/2$ -BPSK, BPSK, QPSK, 16/64/256-QAM

Each block provides an interface and architecture for HDL code generation and hardware deployment.

## **1536-Point FFT Example: Implement a 1536-point FFT using a single 512-point FFT block**

The LTE standard provides multiple transmission bandwidth options ranging from 1.4 MHz to 20 MHz. For 15 MHz transmission bandwidth, the standard requires an FFT length of 1536 for OFDM modulation and demodulation. The HDL Implementation of Streaming 1536-point FFT example shows how to implement a 1536-point FFT using a single 512-point FFT HDL Optimized block.

## **Variable sample-rate input for OFDM Demodulator block**

The OFDM Demodulator block now has a parameter that enables you to choose between applying 30.72 MHz input samples for all NDLRBs or applying input samples at a sample rate determined by the NDLRB of each cell.



---

## **TDD Support for SIB Recovery Reference Application: Recover System Information Block type 1 (SIB1) data from time-division duplex LTE signals**

The LTE HDL SIB1 Recovery reference application now shows how to decode SIB1 data for LTE networks that use either TDD or FDD. The design supports HDL code generation with HDL Coder and is ready for deployment to hardware.

### **MIB recovery with strongest cell detection**

Previously, the LTE HDL MIB Recovery reference application selected the first detected cell that met a threshold. This reference application now uses a longer synchronization window and tests several cells before choosing the strongest cell ID to decode.

### **Merge of MIB and MIB TDD reference applications into single model**

The LTE HDL MIB Recovery reference application now integrates the decoding of TDD signals, previously included in the LTE Cell Search and MIB Recovery Using TDD Mode reference application.

### **Separation of LTE PSS and SSS detection into LTE Cell Search reference application**

The PSS and SSS detection subsystems used by the LTE HDL MIB Recovery and LTE HDL SIB1 Recovery reference applications are now described in a separate reference application. See LTE HDL Cell Search.

### **Accelerate Measurement of Bit-Error-Rate (BER)**

The Accelerate BER Measurement for LTE HDL Turbo Decoder example shows how to speed up measurement of the bit-error-rate (BER) for the LTE HDL Toolbox Turbo Decoder by using `parSim` to parallelize the simulations across EbNo points.



# R2018b

---

**Version: 1.2**

**New Features**

**Bug Fixes**

## **SIB1 Reference Application: Implement an LTE System Information Block type 1 (SIB1) recovery subsystem on your FPGA or ASIC**

The LTE HDL SIB1 Recovery Example reference application continues the LTE receiver design from the LTE HDL Cell Search and MIB Recovery reference application to perform SIB1 decoding. This design supports HDL code generation with HDL Coder and is ready for deployment to hardware.

## **Viterbi Decoder and Depuncturer Blocks: Decode convolutionally encoded bit streams using the Viterbi algorithm with puncturing, terminated, and truncated modes**

The Viterbi Decoder block supports continuous, terminated, and truncated modes using hardware-friendly control signals. This block provides an optional reset port for use with continuous mode and supports punctured code rates by providing an erasure port. The Depuncturer block accepts a puncture vector as either a port or property and provides an erasure output signal.

Both blocks support HDL code generation with HDL Coder.

## **Reset port for OFDM Demodulator block**

The OFDM Demodulator block now has a parameter to enable a reset input port.

## **Variable-size FFT example with arbitrary input pattern**

The HDL Implementation of a Variable-Size FFT example now includes a second model that shows how to handle input data that does not conform to the input valid pattern expected by the original model. The new model shows how to process data that is continuously valid and padded with zeros at the end of each symbol.

# R2018a

---

**Version: 1.1**

**New Features**

**Bug Fixes**

## **TDD Support for MIB Recovery Reference Application: Recover master information block data from time-division duplex LTE signals**

This reference application shows PSS and SSS signal detection, duplex mode detection, and MIB decoding for LTE networks that use TDD or FDD. The design supports HDL code generation with HDL Coder and is ready for deployment to hardware. See [LTE Cell Search and MIB Recovery Using TDD Mode](#).

## **5G Filtered OFDM Modulation Example: Implement a filtered orthogonal frequency division multiplexing transmitter in hardware**

This example implements an F-OFDM modulator suitable for use in 5G transmitter designs, and verifies the design using the 5G Library for LTE System Toolbox®. The example shows how to convert from double to fixed-point types and minimize the resource use of the design on an FPGA. See [HDL Code Generation for Filtered OFDM \(F-OFDM\) Transmitter](#).

## **Gold Sequence Generator Block: Generate LTE-specific Gold sequences for channel estimation and descrambling**

The Gold Sequence Generator block returns Gold sequences generated using the polynomial and shift length specified by LTE standard TS 36.212. LTE systems use a Gold sequence generator for reference symbols and for scrambling/descrambling data, such as in MIB and SIB coding and decoding.

## **OFDM Demodulator Block: Demodulate orthogonal frequency division multiplexing symbols according to the LTE standard**

The OFDM Demodulator block implements an algorithm for demodulating LTE signals specified by LTE standard TS 36.212, providing an interface and architecture for HDL code generation and hardware deployment. The block demodulates time-domain OFDM samples and returns the resource grid that is used for cell ID detection, MIB recovery, SIB1 recovery, and further decoding.

## **Variable-Size FFT Example: Implement flexible-bandwidth FFT using a single FFT block**

This example shows how to implement an FFT that can be used for different LTE standard bandwidths. The design uses a controller around an FFT HDL Optimized block. See [HDL Implementation of a Variable-Size FFT](#).

# R2017b

---

**Version: 1.0**

**New Features**

## Standard-compliant LTE Simulink blocks

LTE HDL Toolbox blocks implement the CRC, Turbo, and Convolutional encoding and decoding algorithms specified by LTE standard TS 36.212, using hardware-friendly architectures and interfaces. For the list of algorithms provided in this product, see HDL-Optimized System Design.

## Downlink detector and MIB recovery reference application

The product includes a reference application that performs PSS and SSS signal detection and MIB decoding. This design supports HDL code generation with HDL Coder and is ready for deployment to hardware. See LTE Cell Search and MIB Recovery.

## HDL code generation support

The encoding and decoding blocks in LTE HDL Toolbox support HDL code generation. For the list of blocks, see HDL-Optimized System Design. To generate HDL code from these designs, you must have an HDL Coder license. HDL Coder also enables you to generate scripts and test benches for use with third-party HDL simulators.

## Frame-to-sample and sample-to-frame conversions

Hardware designs must use a streaming data interface, while LTE Toolbox™ scripts operate on frame-based data. LTE HDL Toolbox provides frame-to-sample and sample-to-frame conversion capability that enables verification of hardware algorithms against LTE Toolbox designs. For details of the streaming interface, see Streaming Sample Interface.

Convert framed data in MATLAB® to a sample stream using `ltehdlFramesToSamples`, then import the samples to Simulink® for HDL-compatible design. For an example of this workflow, see Verify Turbo Decoder with Streaming Data from MATLAB.

Alternatively, import framed data from MATLAB to Simulink and convert the frames to a sample stream using the Frame to Samples block. For an example of this workflow, see Verify Turbo Decoder with Framed Data from MATLAB.

## FPGA-in-the-loop verification on Xilinx or Intel boards

If you have an HDL Verifier™ license, you can use the FPGA-in-the-loop (FIL) feature to verify your HDL design against models in MATLAB and Simulink, while your design runs on an FPGA board. The FIL blocks provide efficiency improvements for streaming data across the interface between Simulink and the FPGA board. See FPGA-in-the-Loop.

## Algorithm prototyping on Xilinx Zynq-based hardware

The Communications Toolbox™ Support Package for Xilinx® Zynq®-Based Radio enables you to design, prototype, and verify practical wireless communications systems on Xilinx Zynq-based radio hardware. LTE HDL Toolbox designs have compatible interfaces to fit into the SDR support package reference design. See Prototype LTE Algorithms on Hardware.