A Myriad of Wireless Standards

…and growing complexity
From Design to Prototype and beyond

- A **common** design environment across multiple teams
- Target **off-the-shelf** hardware for **prototype** development
- Minimize time to market
Modeling Wireless Standards with MATLAB & Simulink
Typical Use Cases

Golden Reference for Verification
Verify in-house PHY models

End-to-End Link-Level Simulation
How do design choices affect system performance?

Signal Generation and Analysis
Test with live data

Signal Information Recovery
Decode real-world signals
WLAN System Toolbox

Supported Standards

802.11ah
- S1G – Sub 1 GHz
  - 900 MHz
  - long range
  - low power
  - IoT

802.11b/g/n
- VHT – Very High Throughput
- HT – High Throughput
- Non-HT
  - 2.4/5 GHz
  - medium range
  - general access

802.11a/ac/ax
- DMG – Directional Multigigabit
  - 60 GHz,
  - short range
  - high throughput wireless video

802.11ad

Non-HT
- 5.9 GHz
- V2V/V2X

MATLAB EXPO 2018
LTE System Toolbox
Granularity

PDSCH Example

Low level functions

Scrambling
Scrambling: ltePDSCHPRBS

Modulation
Modulation: lteSymbolModulate
Demodulation: lteSymbolDemodulate

Layer Mapper
Layer mapping: lteLayerMap
Layer demapping: lteLayerDemap

Precoding
Precoding: lte3LPrecode
Deprecoding: lte3LDeprecode

PMI Set
Precoding matrix indication:
ltePMISelect
ltePMIInfo
lteCSICodebook

Codeword

Scrambling
Scrambling: ltePDSCHPRBS

Modulation
Modulation: lteSymbolModulate
Demodulation: lteSymbolDemodulate

Layer Mapper
Layer mapping: lteLayerMap
Layer demapping: lteLayerDemap

Precoding
Precoding: lte3LPrecode
Deprecoding: lte3LDeprecode

Complete PDSCH Processing:
Encoding:
ltePDSCH
Decoding:
ltePDSCHDecode

Mid-level functions

Resource Mapping
Resource indices:
ltePDSCHIndices

Resource Mapping
Resource indices:
ltePDSCHIndices

Resource Mapping
Resource indices:
ltePDSCHIndices

Resource Mapping
Resource indices:
ltePDSCHIndices

MATLAB EXPO 2018
Wireless Modeling Challenges

Baseband DSP development
- Is my implementation correct?
- Evaluate link performance (algorithm)?

Antenna array design and evaluation
- Element coupling
- Edge effects

Explore beamforming trade-offs
- Baseband, analogue or hybrid
- Simulate capabilities and limitations
- Trade-off ADCs vs RF components

Impact of RF impairments
- Frequency dependency
- Non-linearities
- Mismatches and coupling

MATLAB EXPO 2018
Wireless design challenges

The Model Based Design Advantage

System Architecture
Mixed-Signal Hardware
Antenna Design

DSP Algorithms
Digital Hardware

Software Development
RF Design

Executable Specifications
Continuous Test and Verification
Automatic Code Generation

Design with Simulation

Requires 7 different skills to be successful!

^ at least
Modeling 802.11ad – Including Beamforming

- Uniform linear array of 4 elements (TX & RX)
- MIMO channel with 6 scatterers
- PER and EVM for 802.11ad link
Modelling 802.11ad Beamforming
Working with Real Signals

Going beyond simulation
## Supported Hardware for Radio Connectivity

### Signal Generator and Analyser

- **Keysight, R&S, NI, Tektronix, …**
- **High quality RF front end**
- **Wide frequency range, high bandwidth**

### SDR

- **USRP, PLUTO, Zynq, …**
- **Customizable RF front end**
- **Sizable FPGA for targeting designs**

### Ultra low-cost SDR

- **RTL-SDR, …**
- **Low bandwidth**
- **Receive only**

---

---
Radio I/O

Stream Over the Air signals – Explore spectrum
Modelling MIB Decoding

- Emulate the User Equipment using a SDR
- Implement LTE Cell Search procedure
- Physical layer processing:
  - Cell search
  - Time and frequency offset estimation and correction
  - OFDM demodulation
  - Channel estimation and equalization
  - PBCH Demodulation
  - BCH Decoding
  - MIB Parsing

3GPP TS36.331 Section 6.2.2
Modelling MIB Decoding

- **Golden reference**
  - Ensure my implementation is correct
  - E.g. PSS/SSS sequence for cell search

- **Algorithm development**
  - Link-level simulation to analyse receiver IP performance
  - E.g. channel estimation

- **Verification**
  - Capture and decode over-the-air waveforms offline
  - Verify behaviour before moving to HW

3GPP TS36.331 Section 6.2.2
DEMO LTE Scanner

SDR

MATLAB

LTE System Toolbox™
LTE Cell Scanner

Reference Signal Measurement vs. Frequency

CellID: 456
NDLRB: 50

RSRQ: -21.81 dB
RSRP: 35.96 dBm
RSSI: 65.61 dBm

Cell settings from MIB decoding:
- Frequency: 806 MHz
- DuplexMode: PDD
- CyclicPrefix: Normal
- NDLRB: 50
- NCellID: 456
- NSubframe: 0
- CellRefP: 2
- PHICHDuration: Normal
- Ng: One
- NFrame: 181

Choose recovered cell information:
- 806 MHz - Cell ID: 456

PDCH settings from DCI Decoding:
- RNTI: 
- PRBSSet: 
- Nlayers: 
- Modulation: 
- RV: 
- TxScheme:
MATLAB
- Large data sets
- Explore mathematics
- Data visualization

Simulink
- Model HW Parallel architectures
- Simulation
- Code Generation Capabilities

Targeting FPGA and ASIC
- Streaming design
- Implementation detail
- Architectural specification
- Verification
Typical Workflow

Modeling & Simulation

Radio I/O
Streaming real-world data

Deploy on hardware

Standalone operation
Zynq SDR - Hardware Support Package

Embedded Coder

ARM
- Algorithm C
- Driver

HDL Coder

FPGA IP
- Algorithm HDL
- AXI interface

Placeholder for C/C++ application

AXI Interface R/W

Parameters

Monitors

Tx Data

Rx Data

RF Rx Block

RF Tx Block

Placeholder for FPGA User IP

User Logic

AXI-Lite

AXI DMA

Tx DMA I/Q

Rx DMA I/Q

Rx baseband I/Q

Tx baseband I/Q

RF HDL IP

RF Card

RF In

RF Out

Zynq board

ARM (PS)

External Mode TCP/IP

Data/control Source

Display/Scope etc

Simulink model

MATLAB EXPO 2018
Targeting an algorithm to the FPGA and ARM

Run on ARM Processing System

Run on Programmable Logic
LTE Cell Scanner Example: Algorithm

- Model Algorithm
- Generate Bitstream
- SW Interface Model
- Run on Hardware
LTE Cell Scanner Example: Generation

- Model Algorithm
- Generate Bitstream
- SW Interface Model
- Run on Hardware
HDL Advisor

Step by Step Assistant: Set Target

1.3. Set Target Interface

Analysis (Triggers Update Diagram)
Set target interface for HDL code generation
Input Parameters
Processor/FPGA synchronization: Free running

Target platform interface table

<table>
<thead>
<tr>
<th>Port Name</th>
<th>Port Type</th>
<th>Data Type</th>
<th>Target Platform Interfaces</th>
<th>Bit Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>RxDataI_In</td>
<td>Inport</td>
<td>sfix16_E</td>
<td>Rx data I1 In [0:15]</td>
<td>[0:15]</td>
</tr>
<tr>
<td>RxDataQ_In</td>
<td>Inport</td>
<td>sfix16_E</td>
<td>Rx data Q1 In [0:15]</td>
<td>[0:15]</td>
</tr>
<tr>
<td>RxDataValid_In</td>
<td>Inport</td>
<td>boolean</td>
<td>Rx data Valid In</td>
<td>[0]</td>
</tr>
<tr>
<td>start</td>
<td>Inport</td>
<td>boolean</td>
<td>AXI4-Lite</td>
<td>x’104^*</td>
</tr>
<tr>
<td>externalDataSel</td>
<td>Inport</td>
<td>boolean</td>
<td>AXI4-Lite</td>
<td>x’108^*</td>
</tr>
<tr>
<td>RxDataI_Out</td>
<td>Outport</td>
<td>int16</td>
<td>Rx data I1 Out [0:15]</td>
<td>[0:15]</td>
</tr>
<tr>
<td>RxDataQ_Out</td>
<td>Outport</td>
<td>int16</td>
<td>Rx data Q1 Out [0:15]</td>
<td>[0:15]</td>
</tr>
</tbody>
</table>
HDL Advisor
Step by Step Assistant: Check Model

- HDL Workflow Advisor
  - 1. Set Target
  - 2. Prepare Model For HDL Code Generation
    - 2.1. Check Global Settings
    - 2.2. Check Algebraic Loops
    - 2.3. Check Block Compatibility
    - 2.4. Check Sample Times
  - 3. HDL Code Generation
  - 4. Embedded System Integration

2. Prepare Model For HDL Code Generation

- Analysis
- Check and Prepare Model for HDL code generation
- Run All
  - Show report after run

Report
- Report: ...eport_1332.html
- Save As...
- Date/Time: 25-May-2018 17:44:44
- Summary: Pass: 4 Fail: 0 Warning: 0

MATLAB EXPO 2018
HDL Advisor

Step by Step Assistant: Generate HDL

3.2. Generate RTL Code and IP Core
Analysis (Triggers Update Diagram)
Generate RTL code and IP core for embedded system
Input Parameters

- IP core name: LTE_MIB_HDL_ip
- IP core version: 1.0
- IP core folder: C:\Temp\HDLPipcore\LTE_MIB_HDL_ip_v1_0
- IP repository:
- Additional source files:
- Generate IP core report

Run This Task
Result: Passed
Generated HDL Code

Navigating Code and Viewing Reports
HDL Advisor

Step by Step Assistant: SW Model & Bitstream

Zynq Radio Software Interface Model: zynqRadioHWSWLTEMIBDetectorAD9361AD9364SL

This model contains blocks that can be used for software generation.

Generated by HDL Workflow Advisor on 25-May-2018 18:19:26
Software Interface

Runs on ARM processor

- Model Algorithm
- Generate Bitstream
- SW Interface Model
- Run on Hardware
Targeting Workflow

Review

1. Setup reference design
2. Generate HDL code
3. Generate Vivado project
4. Generate SW models
5. Generate bitstream
6. Load bitstream
7. Configure SW model
8. Generate SW application
9. Run on hardware
Run on Hardware
Extending Standards

**LTE to 5G**

- **PDSCH generation and mapping**
- **waveform generation: OFDM**
- **LTE channel model**
- **synchronization**
- **OFDM demodulation**
- **channel estimation**
- **PDSCH decoding**

**5G channel models**

- WOLA-OFDM, F-OFDM
- W-OFDM, F-OFDM

**Waveform gen:**
- OFDM, F-OFDM or W-OFDM
- var. subcarrier spacing

**Synchronization channel estimation**
- PDSCH decoding

- **Turbo, LDPC**
- **DL-SCH gen:**
  - Turbo, LDPC
Conclusions

- Wireless Designs starts with MATLAB
  - Prove algorithm and design with simulation and over-the-air signals
  - Generate customizable waveforms to verify conformance to the latest 5G, LTE, and WLAN standards
  - **Automatically generate HDL or C** code for prototyping and implementation without hand-coding
Thank you!