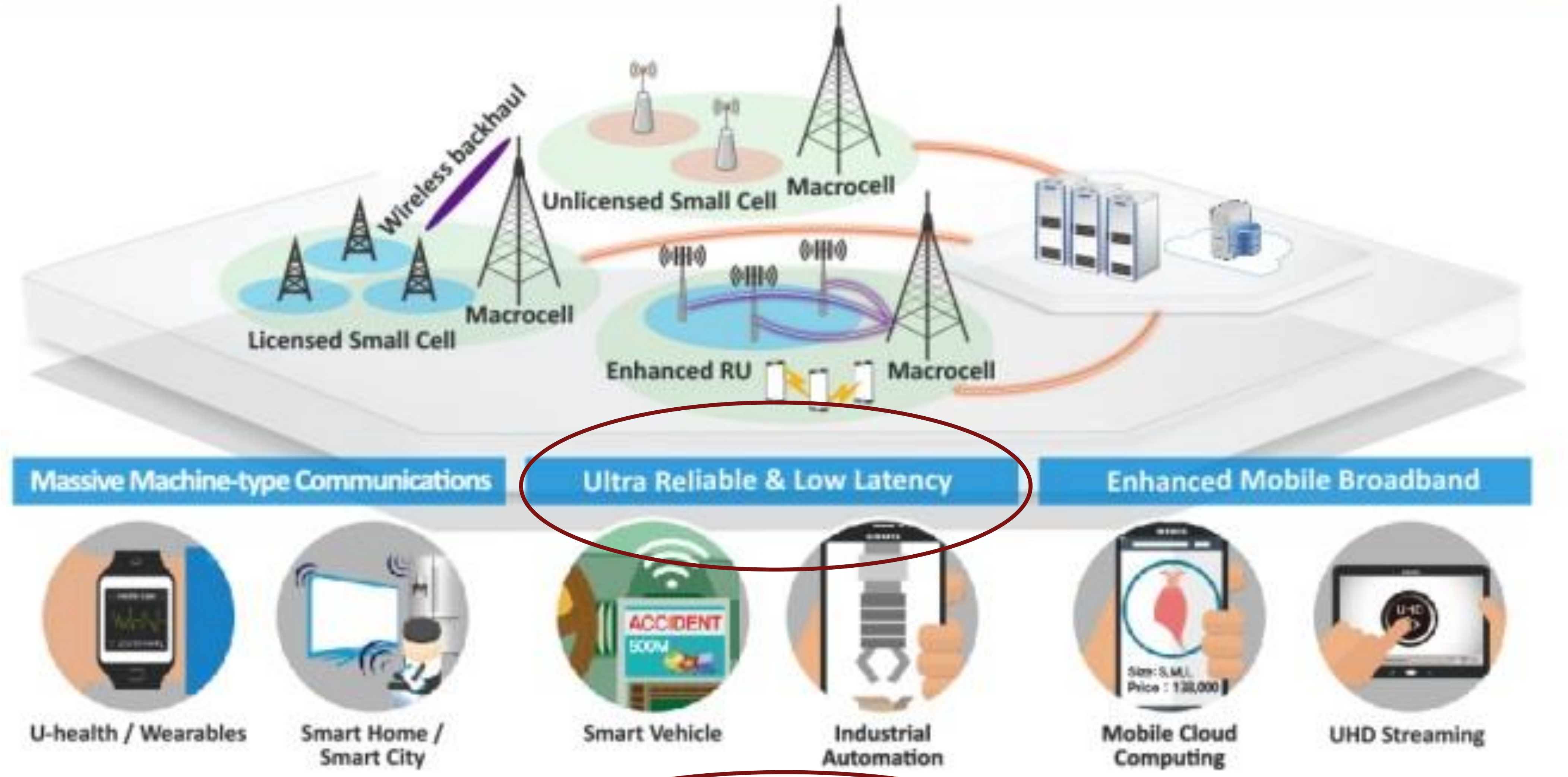
5G & Future of Connected Vehicles

5G Vision and Use Cases



IoT **V2X** **Ultra-Fast Mobile**

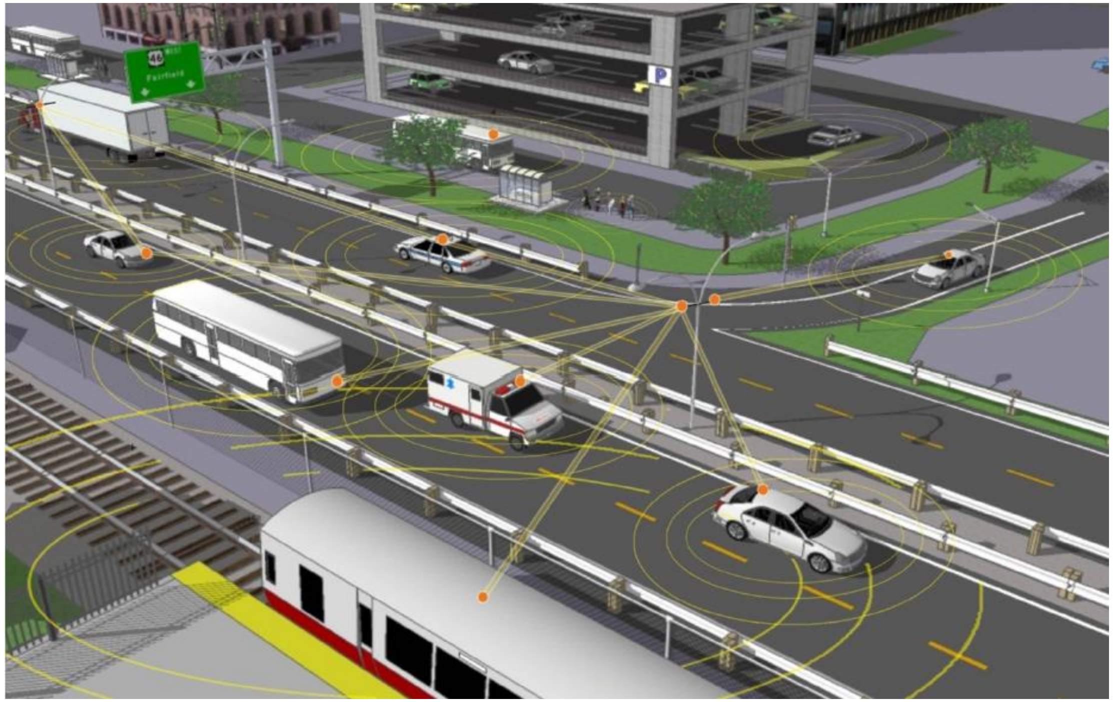
Motivations for V2X

- Safety
  - 33000 death in highway accidents (US DOT, 2012)
  - 5.1 million crashes (US DOT, 2012)
  - Leading cause of death for people of age 11-27
- Mobility & Productivity
  - 5.5 billion hours of traffic delay (per year)
  - 121 billion USD cost of urban congestion (per year)
- Environment
  - 2.9 billion gallons of waste in fossil fuel (per year)
  - 56 billion lbs. of additional emitted CO2

Vehicular Communications

- Vehicle-to-Vehicle (V2V)
- Vehicle-to-Infrastructure (V2I)
- Vehicle-to-Pedestrian (V2P)
- Vehicle-to-Network (V2N)

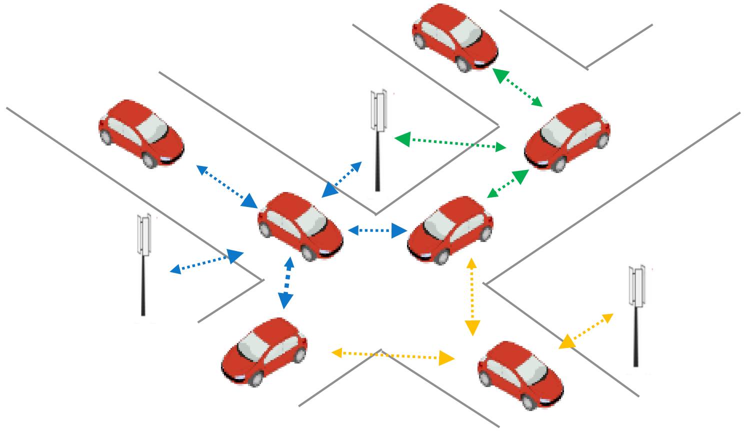
} V2X



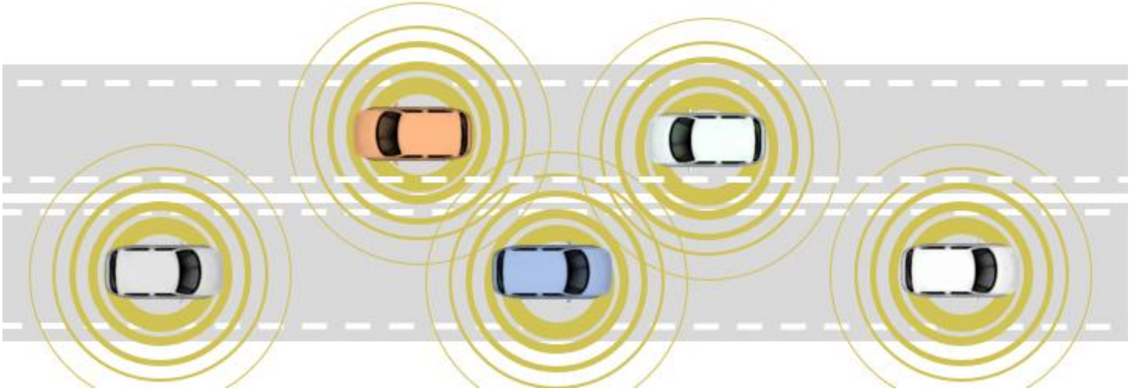
➢ Continuous, high-speed, and authenticable safety data exchange among moving vehicles (V2V) and between vehicles and roadway infrastructure (V2I), pedestrians (V2P) and cellular network (V2N)

V2X Standards & MathWorks Solutions

- DSRC – IEEE 802.11p
- WLAN Toolbox™

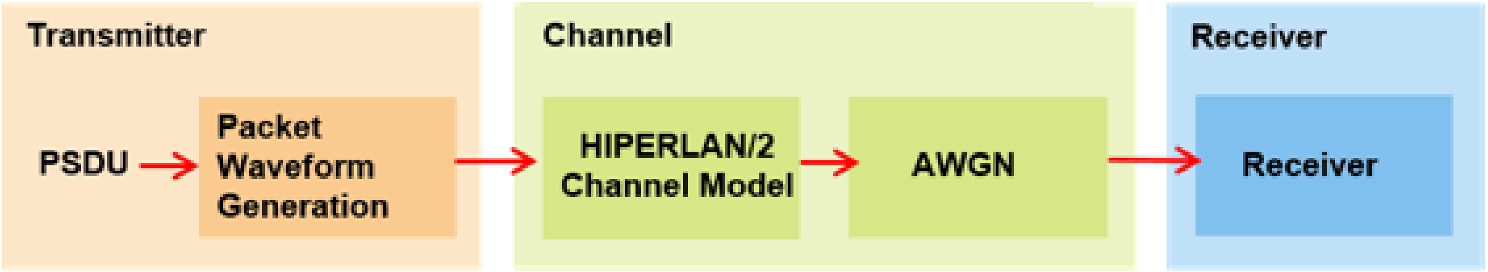


- C-V2X – Release 14 LTE
- LTE Toolbox™

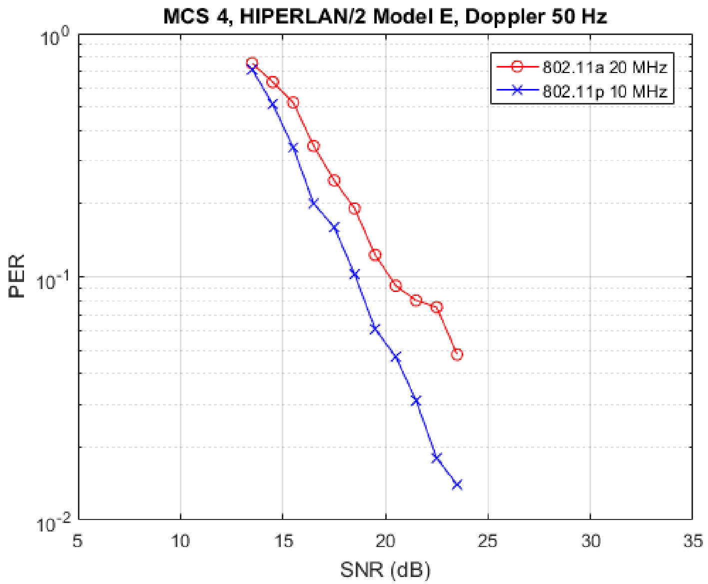


WLAN Toolbox – IEEE 802.11p (DSRC)

- 802.11p and 802.11a Packet Error Rate Simulations



Parameters	802.11a	802.11p
Channel Bandwidth (MHz)	20	10
Bit Rates (Mbps)	6, 9, 12, 18, 24, 36, 48, 54	3, 4.5, 6, 9, 12, 18, 24, 27
OFDM symbol duration (us)	4	8
Guard duration (us)	0.8	1.6
Preamble duration (us)	20	40
Subcarrier Spacing (KHz)	312.5	156.25



https://www.mathworks.com/help/wlan/examples/802-11p-and-802-11a-packet-error-rate-simulations.html

LTE Acceleration into V2V/V2X

- In Release 12, 3GPP defined D2D interface known as **sidelink** within LTE
- In Release 13, study work began on V2V/V2X
- In Release 14, LTE V2X sidelink standard specified

3GPP A GLOBAL INITIATIVE

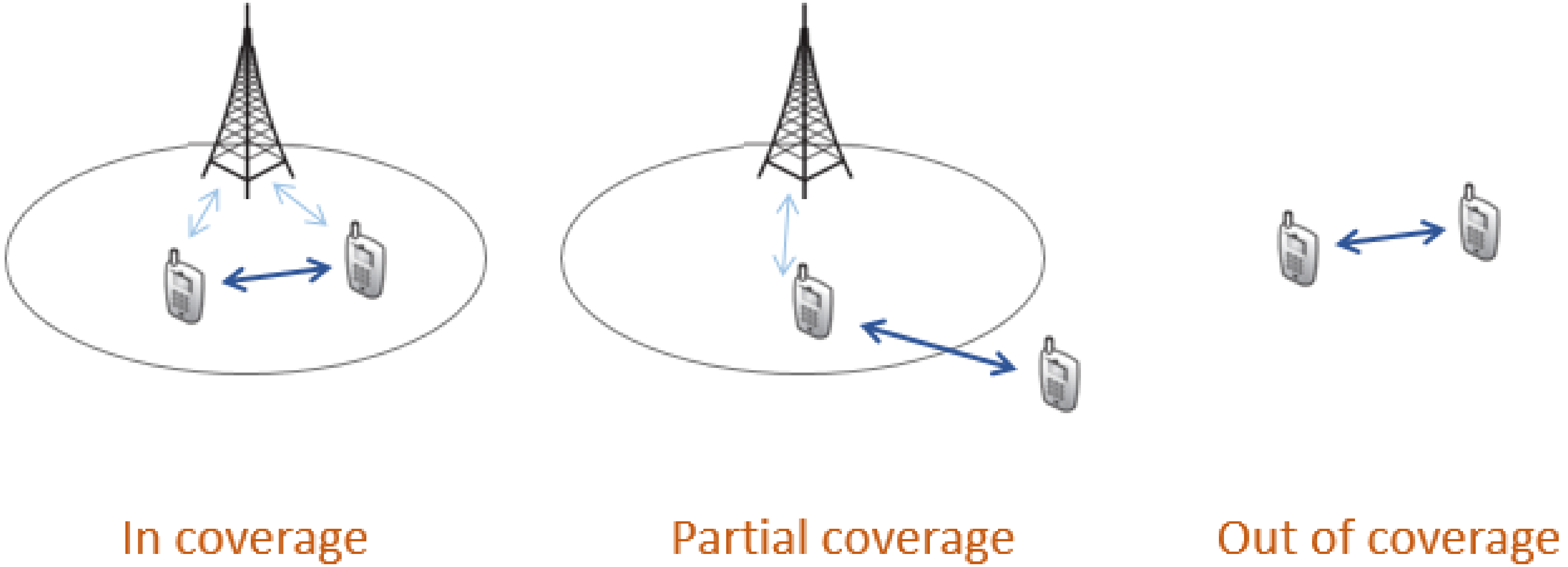
Release 12 sidelink
(03/2015)
R2016b

Release 13 V2V/V2X Study Items
(03/2016)

Release 14 V2X sidelink
(06/2017)
R2017b

What does LTE Sidelink address?

- Ability for UEs to communicate *directly* with or without network assistance

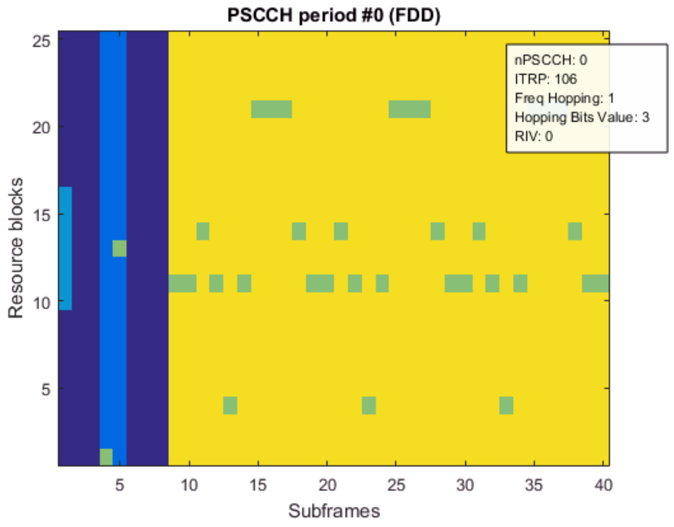


Main application for Direct Communication mode: public safety (firefighters,…)

Sidelink Direct Communications PHY Procedures

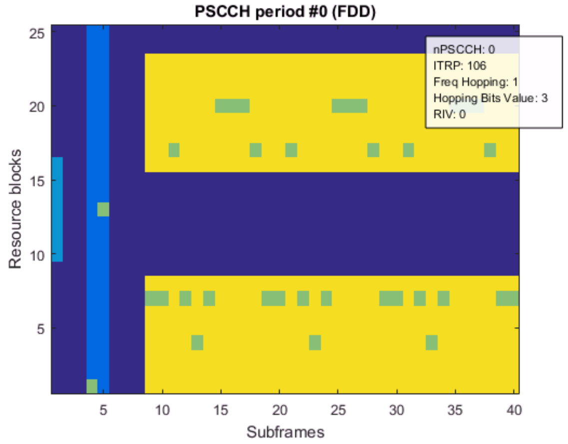
- Uplink transmission resources are used for the sidelink
- The specific sidelink resources in a cell are defined by Tx and Rx pools sent on SIB18 (or pre-configured in the out-of-coverage case)
- The pools occur within recurring time frames known a PSCCH periods
- Transmission resources are selected from a pool using dynamic L1 DCI format 5 signaling in transmission mode 1 (TM1), or at random by the UE in transmission mode 2 (TM2)

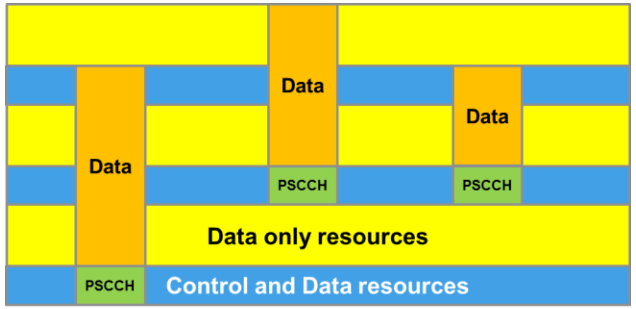
TM1: Network directed
TM2: Autonomously scheduled



LTE System Toolbox Support for Sidelink & V2X

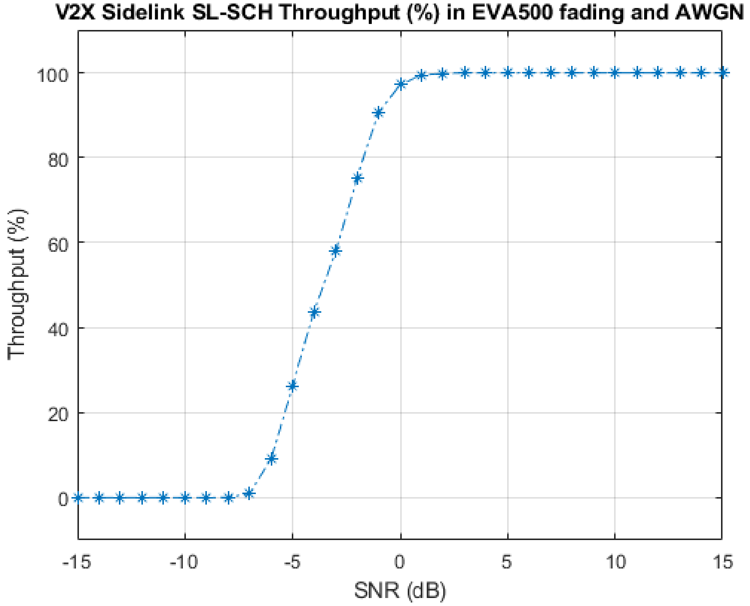
- LTE System Toolbox supports both:
  - Sidelink Direct Communications
  - V2X (since R2017b)





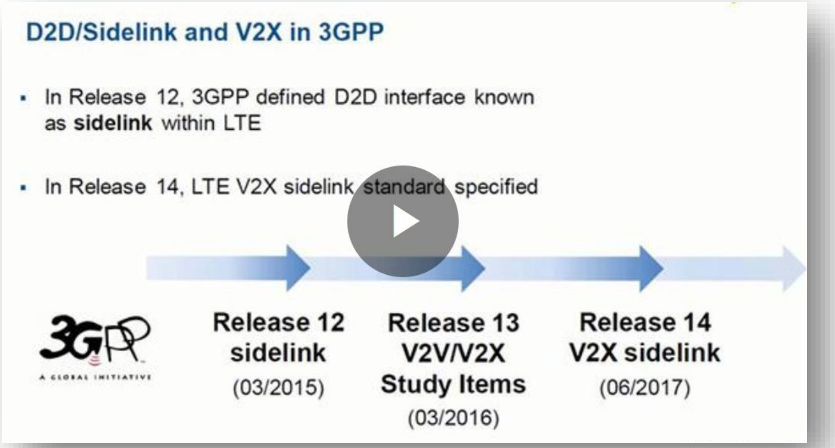
- PSCCH period modeling and waveform generation (PSCCH and PSSCH resource pools, and scheduling assignment period)

- PSCCH and PSSCH link-level throughput in AWGN
- PSCCH BLER for V2X over fading channel



ProSe Sidelink Direct Communications in LTE System Toolbox

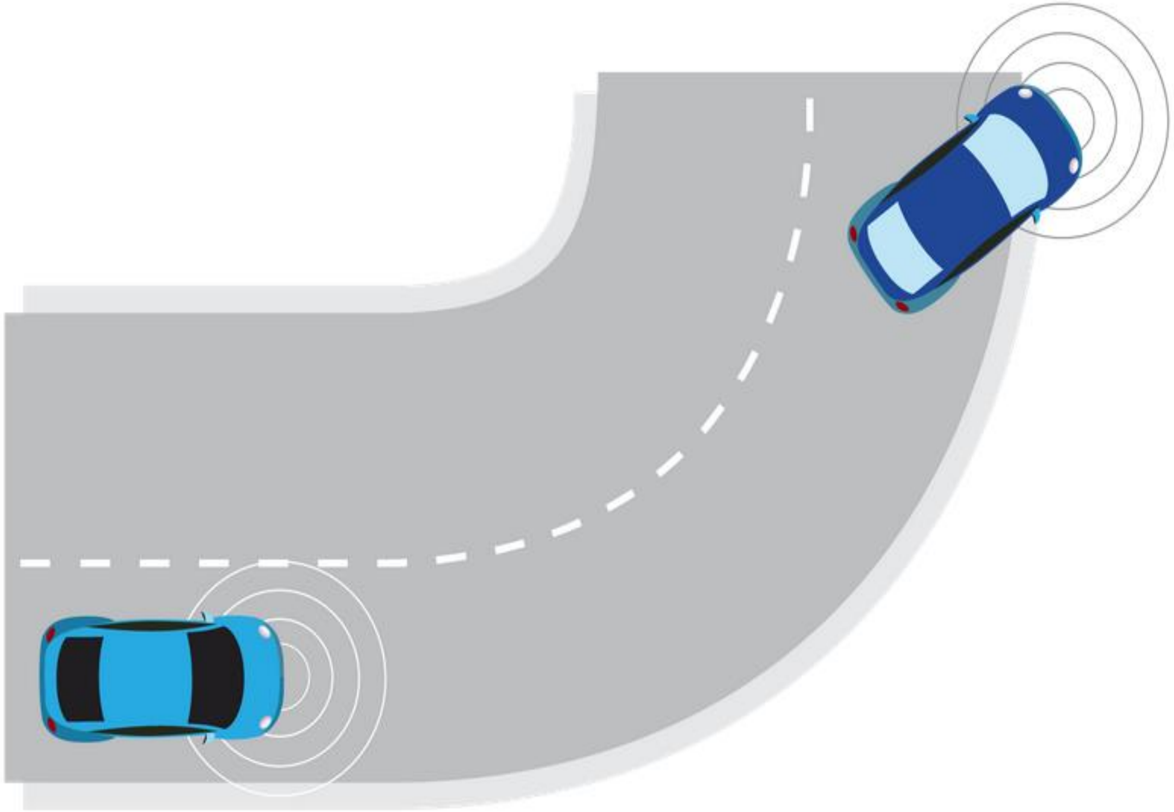
- ProSe sidelink direct communications, transmission modes 1 and 2
- Includes PHY layer transmit/receive functions for FDD/TDD
- Synchronization and broadcast (PSSS/SSSS and MIB-SL, SL-BCH and PSBCH)
- Control (SCI, SCI coding and PSCCH)
- Shared data (SL-SCH and PSSCH)



https://www.mathworks.com/videos/sidelink-and-v2x-modeling-and-simulation-with-lte-toolbox-1534502753891.html

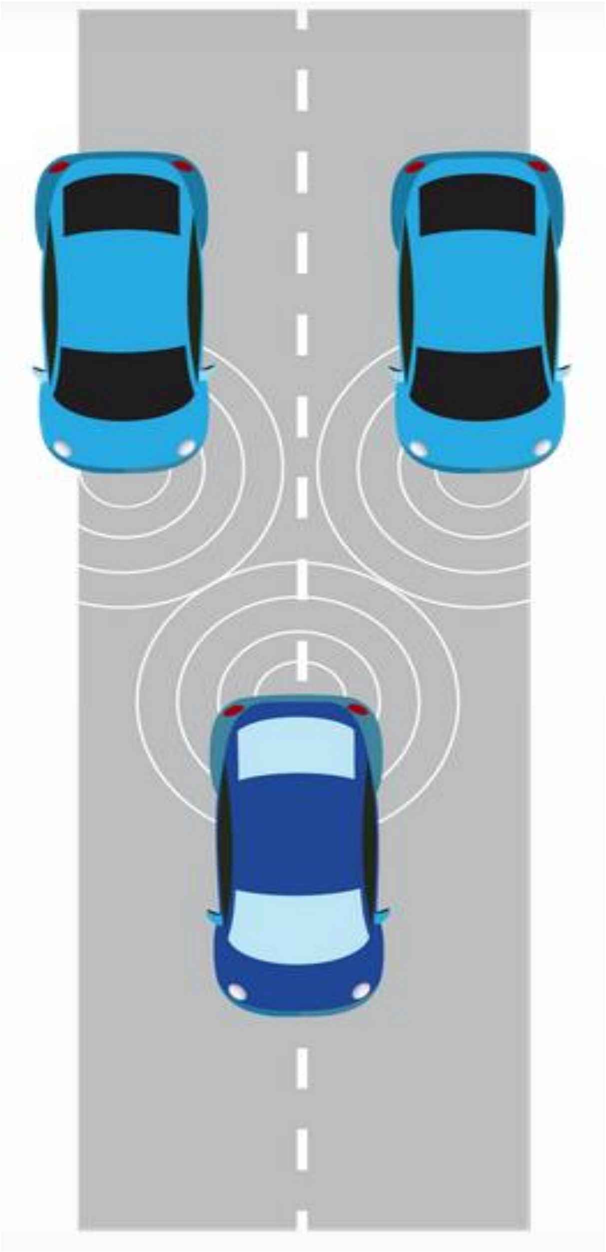
Release 15: Enhanced safety

- Evolution to 5G
- Better Link budget
- Longer range
- More reliability



Release 16: Advanced safety

- Higher throughput
- Higher reliability
- Wide ranging and positioning
- Low latency

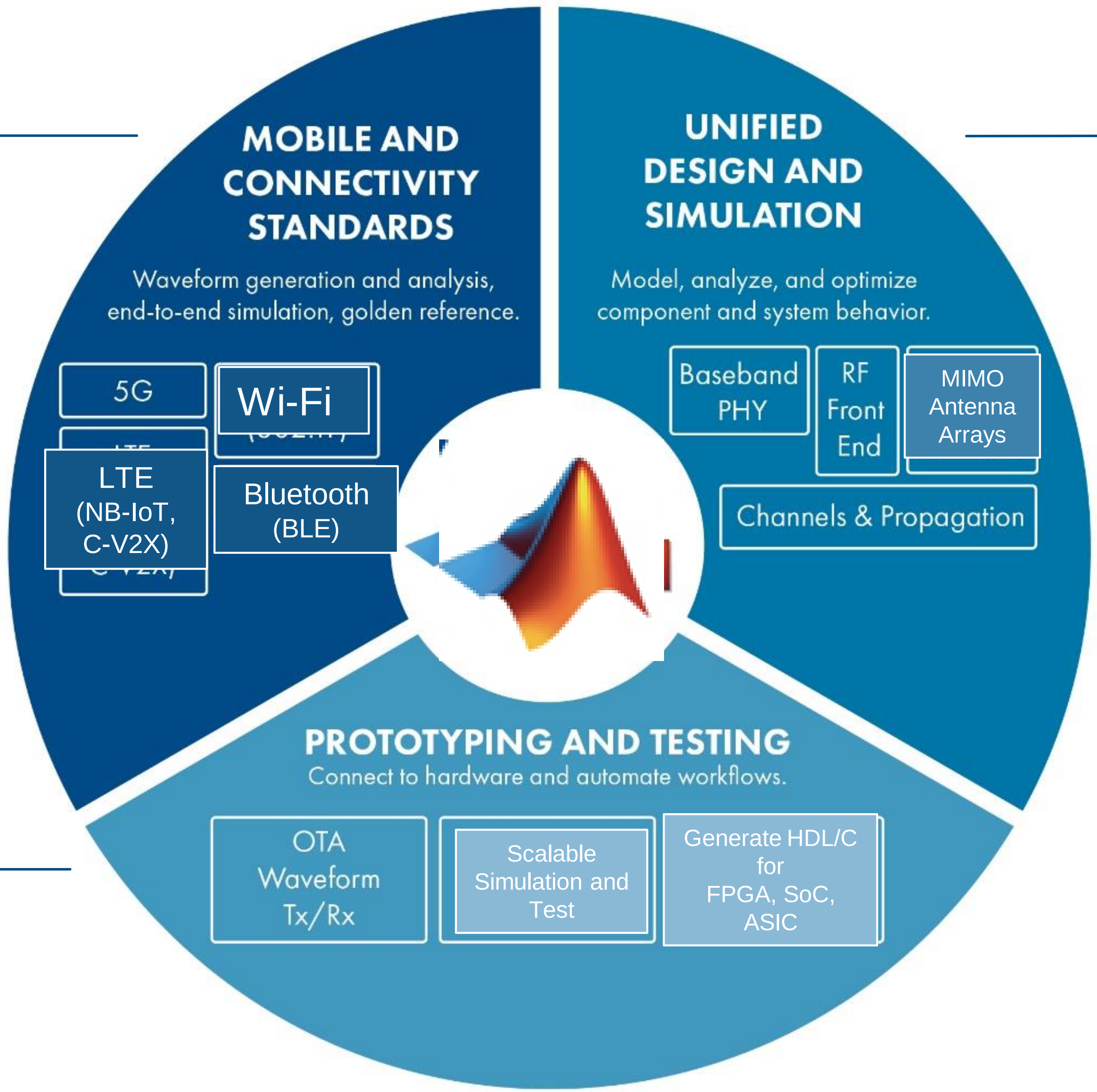


Wireless Communications Development

with MATLAB and Simulink

Related Products

5G Toolbox
LTE Toolbox
WLAN Toolbox
Bluetooth Add-on
(add Communications Toolbox?)



Related Products

Communications Toolbox
RF Toolbox
RF Blockset
Antenna Toolbox
Phased Array System Toolbox

Related Products

Instrument Control Toolbox
Software-Defined Radio Add-ons
MATLAB Coder
MATLAB Compiler
HDL Coder
LTE HDL Toolbox

Learn more: *mathworks.com/wireless*

Model, simulate, design and test 5G systems with MATLAB & 5G Toolbox

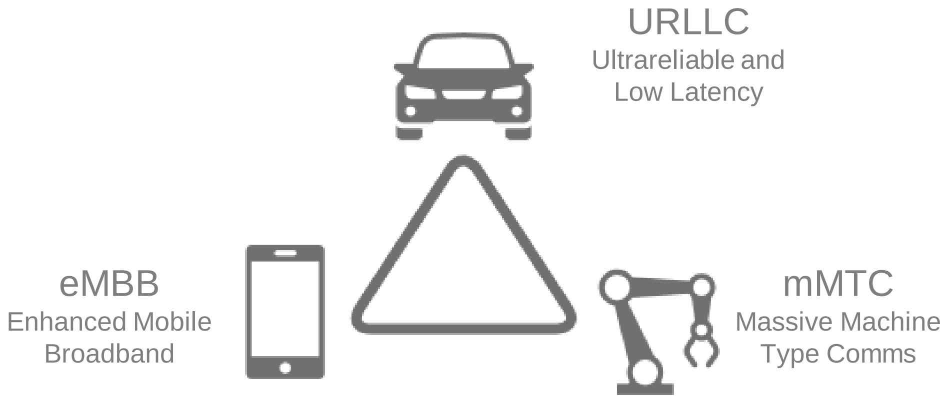
We are investing in 5G and LTE

36.xxx

- LTE-M
- NB-IoT
- Public safety
- C-V2X
- Spectrum sharing

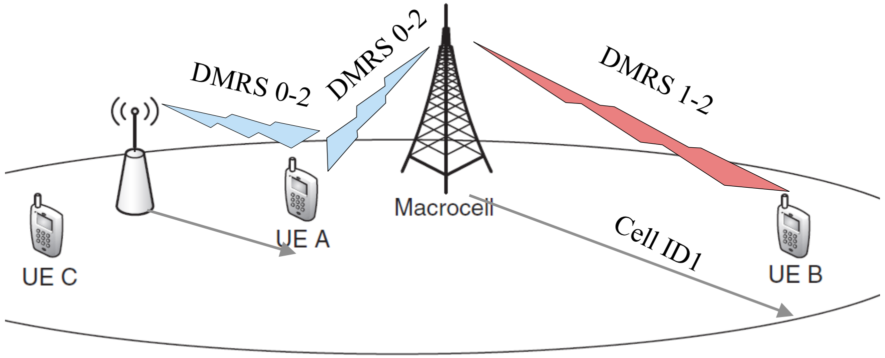
38.xxx

- eMBB
- URLCC
- NR IoT
- NR C-V2X

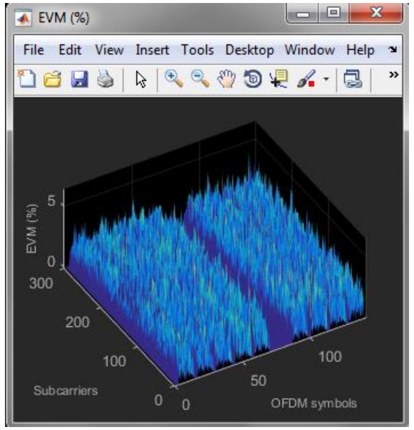


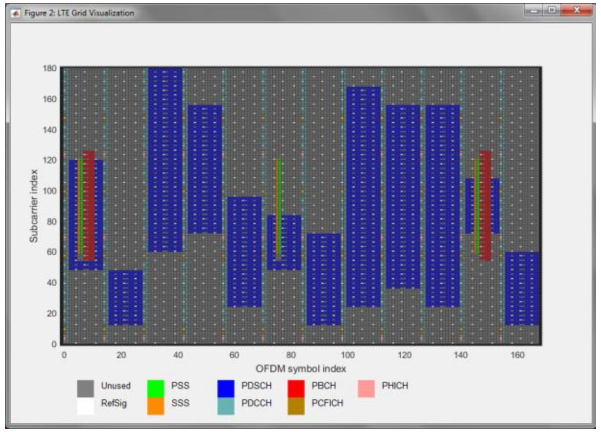
What's LTE System Toolbox?

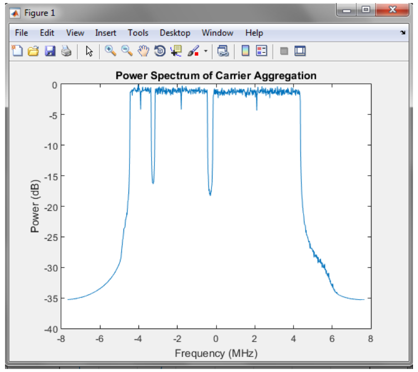
- >200 functions for physical layer (PHY) modeling
- LTE, LTE-Advanced, LTE-Advanced Pro (Rel-8 through Rel-14)
- Scope
  - FDD/TDD
  - Uplink/Downlink/Sidelink
  - Transmitter/Receiver



New in R2017b:
- V2X
- NB-IoT



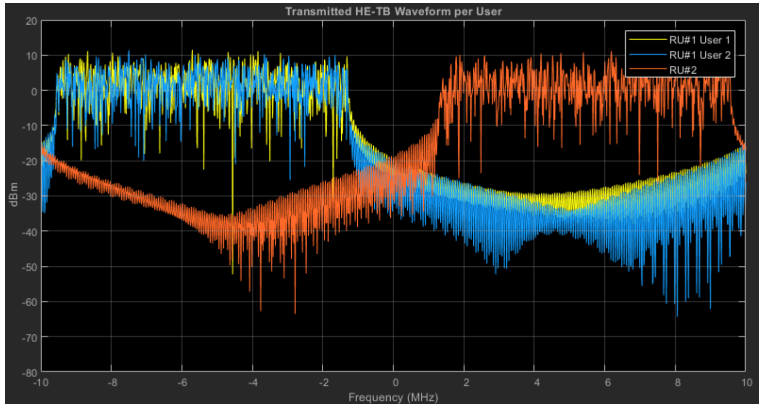


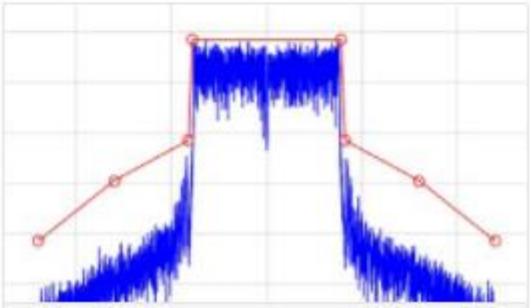


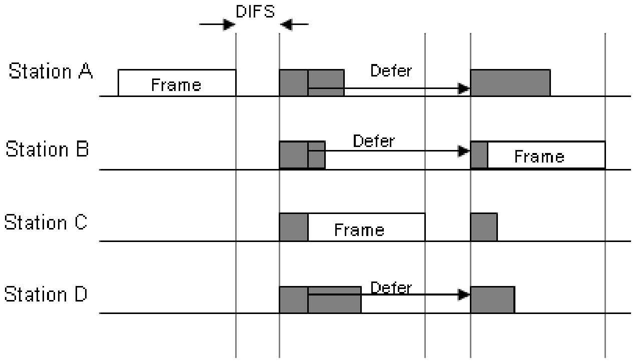
WLAN Toolbox

- Standard-compliant functions for the design, simulation, and analysis of Wi-Fi systems

- Support engineers working across 802.11 layers
…from baseband IQ at the physical layer,
to bits and frames at the MAC layer…



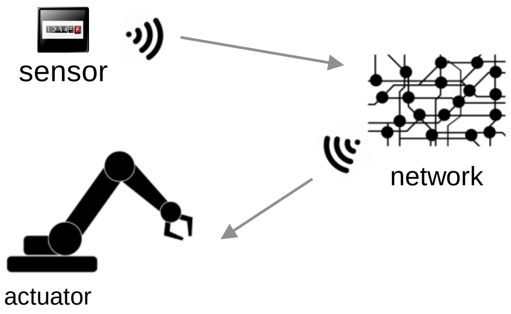




5G Use Cases and Requirements

- eMBB (enhanced Mobile Broadband)
  - High data rates
  - Increased bandwidth efficiency

- mMTC (massive Machine Type Communications)
  - Large number of connections
  - Energy efficiency and low-power operation

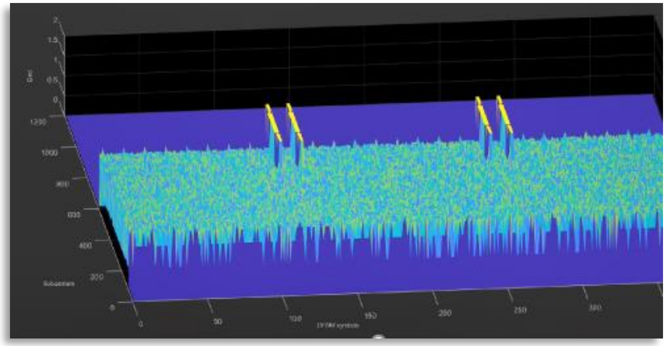


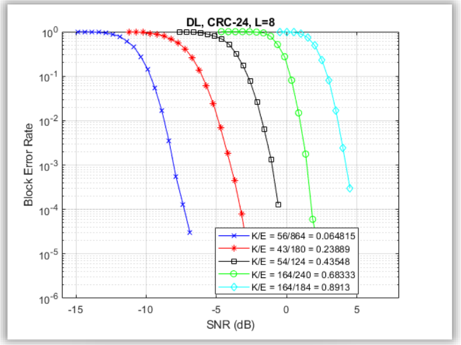
- URLLC (Ultra-Reliable and Low Latency Communications)
  - Low latency

5G vs LTE: Main Physical Layer Differences

	LTE	5G
Use cases	Mobile broadband access (MTC later)	More use cases: eMBB, mMTC, URLLC
Latency	~10 ms	<1 ms
Band	Below 6 GHz	Up to 60 GHz
Bandwidth	Up to 20 MHz	Up to 100 MHz below 6 GHz Up to 400 MHz above 6 GHz
Subcarrier spacing	Fixed	Variable
Freq allocation	UEs need to decode the whole BW	Use of bandwidth parts
“Always on” signals	Used: Cell specific RS, PSS,SSS, PBCH	Avoid always on signals, the only one is the SS block

5G Toolbox applications & customer use-cases





Waveform generation and analysis

- New Radio (NR) subcarrier spacings and frame numerologies

End-to-end link-level simulation

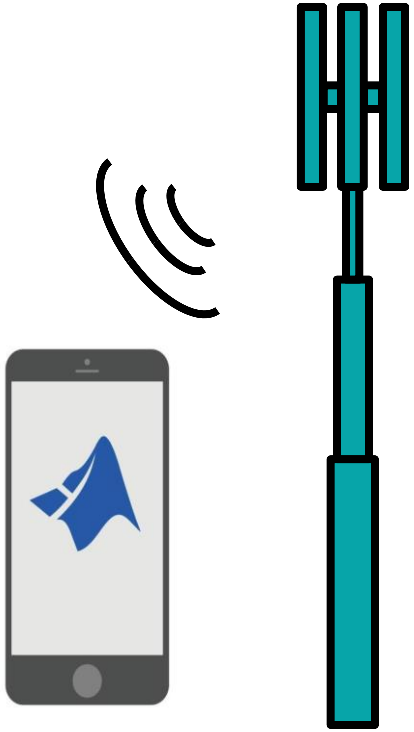
- Transmitter, channel model, and receiver
- Analyze bit error rate (BER), and throughput

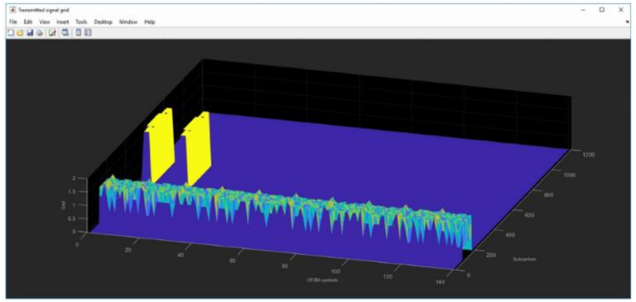
Golden reference design verification

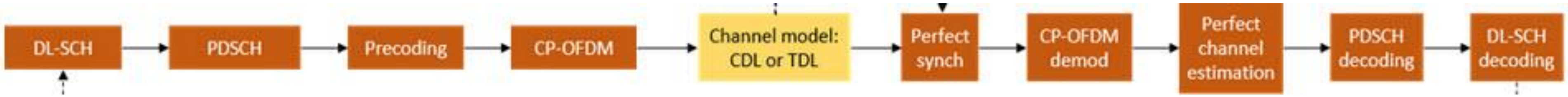
- Customizable and editable algorithms as golden reference for implementation

5G Toolbox

- NR 5G PHY behavioral models in MATLAB
- Transmit and receive for Downlink & Uplink
- TDL and CDL channel models
- Waveform generation
  - Transport channels, physical channels and signals
  - Synchronization bursts
- Reference designs as detailed examples
  - Link-level simulation & throughput measurements
  - Cell search procedures



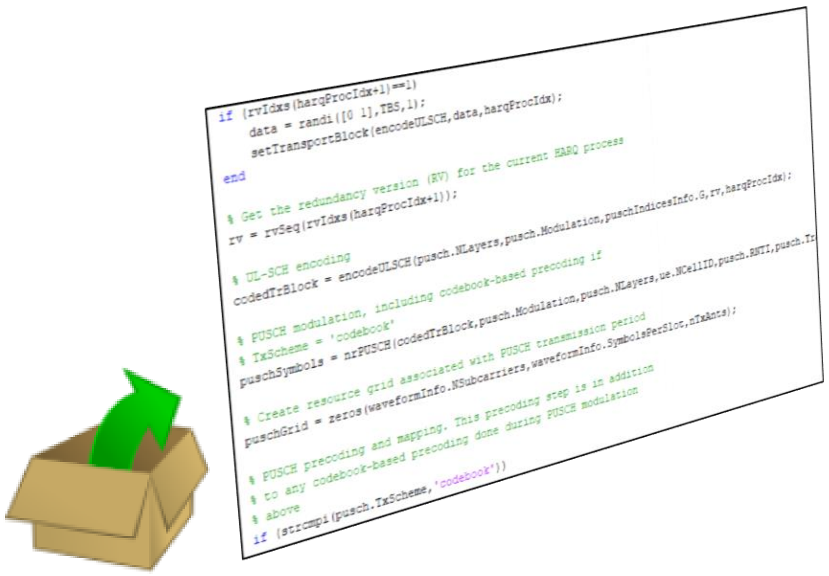




5G Toolbox has open customizable algorithms

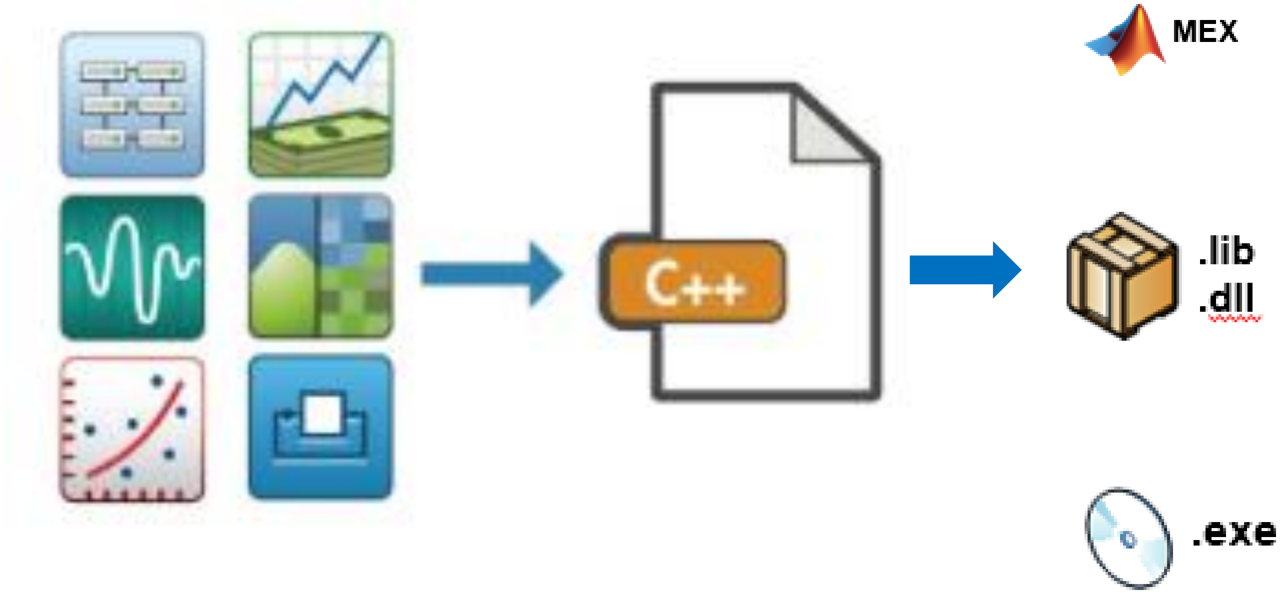
- All functions are

Open, editable, customizable MATLAB code

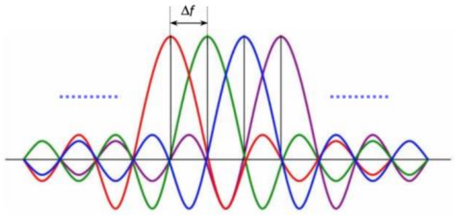


- C/C++ code generation:

Supported with MATLAB Coder

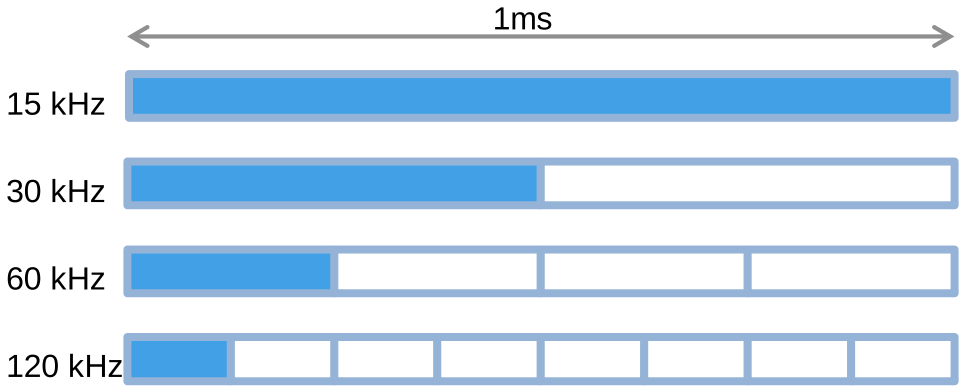


Numerology and Subcarrier Spacing



- Subcarrier spacing can be a power-of-two multiple of 15kHz
- Toolbox supports variable subcarrier spacings
- Waveforms generated by Toolbox can contain a **mix of subcarrier spacings**

μ	Df = 2^{μ} * 15kHz	Slots / ms
0	15	1
1	30	2
2	60	4
3	120	8
4	240	16

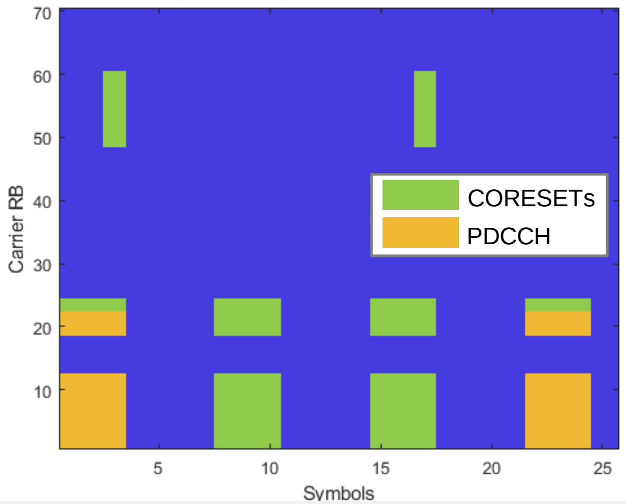


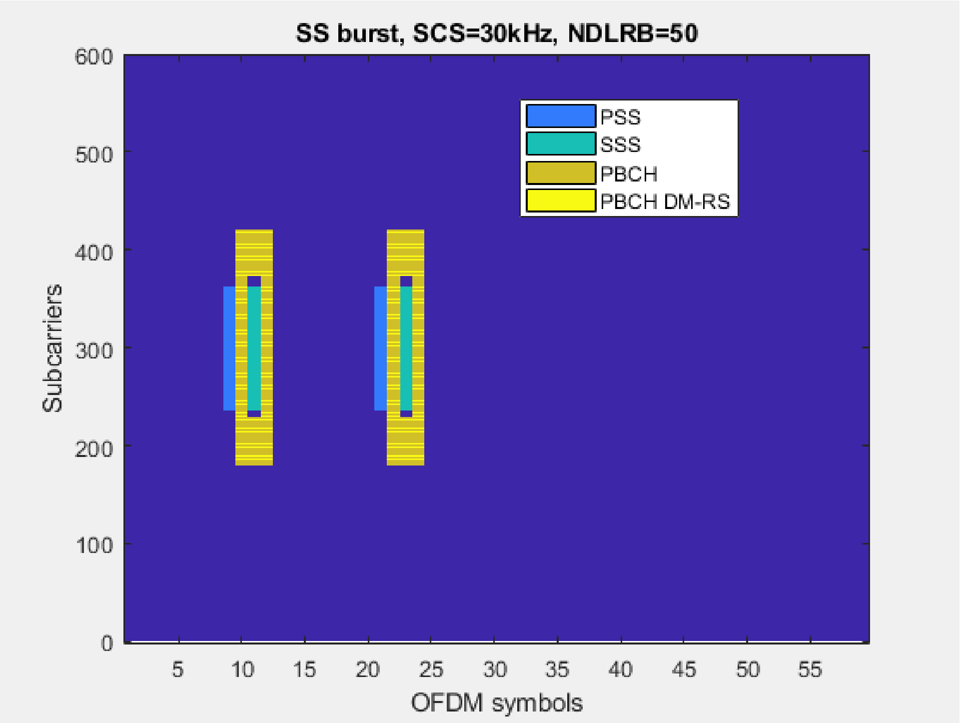
Maximum Supported Bandwidth

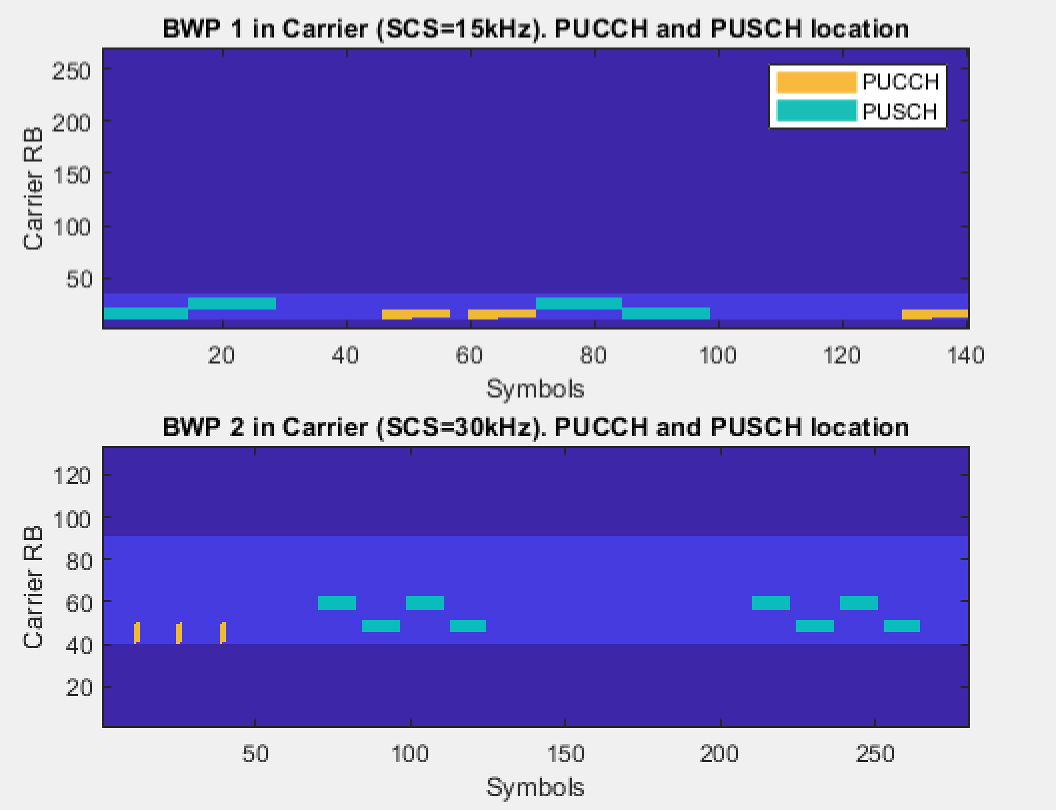
μ	Δf = 2^{μ} * 15kHz	Max NDLRB	nrREs	nrREs * Δf (MHz)
0	15	275	3300	49.50
1	30	275	3300	99
2	60	275	3300	198
3	120	275	3300	396
4	240	138	1656	397.44

Physical layer channels and signals

- Shared, control and broadcast channels
  - PDSCH, PUSCH, PDCCH, PUCCH, PBCH
- Synchronisation and reference signals
  - PSS, SSS, DM-RS

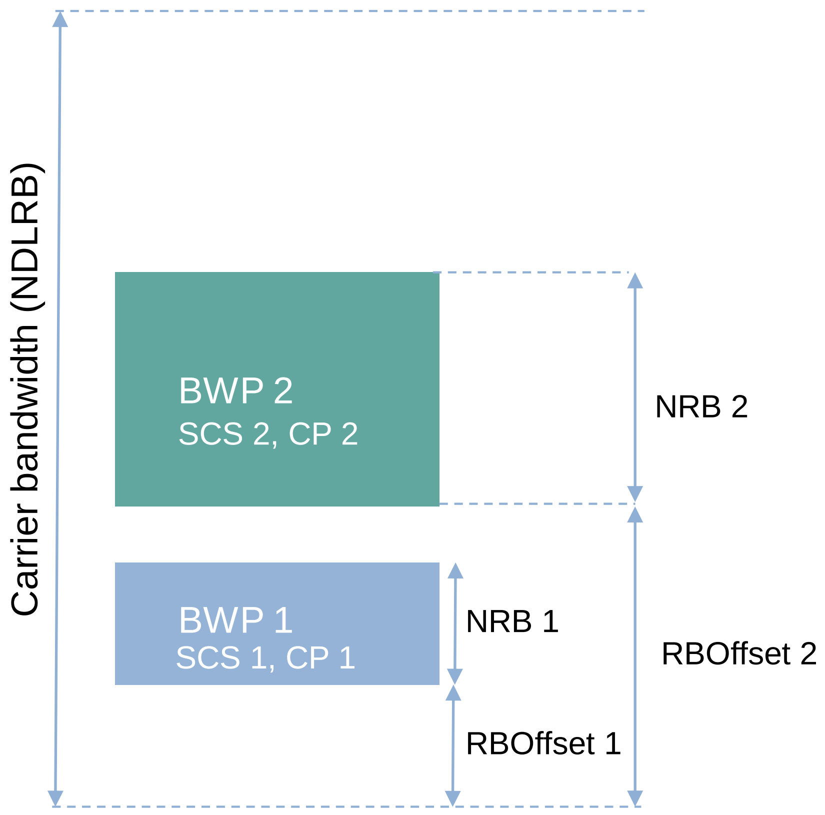




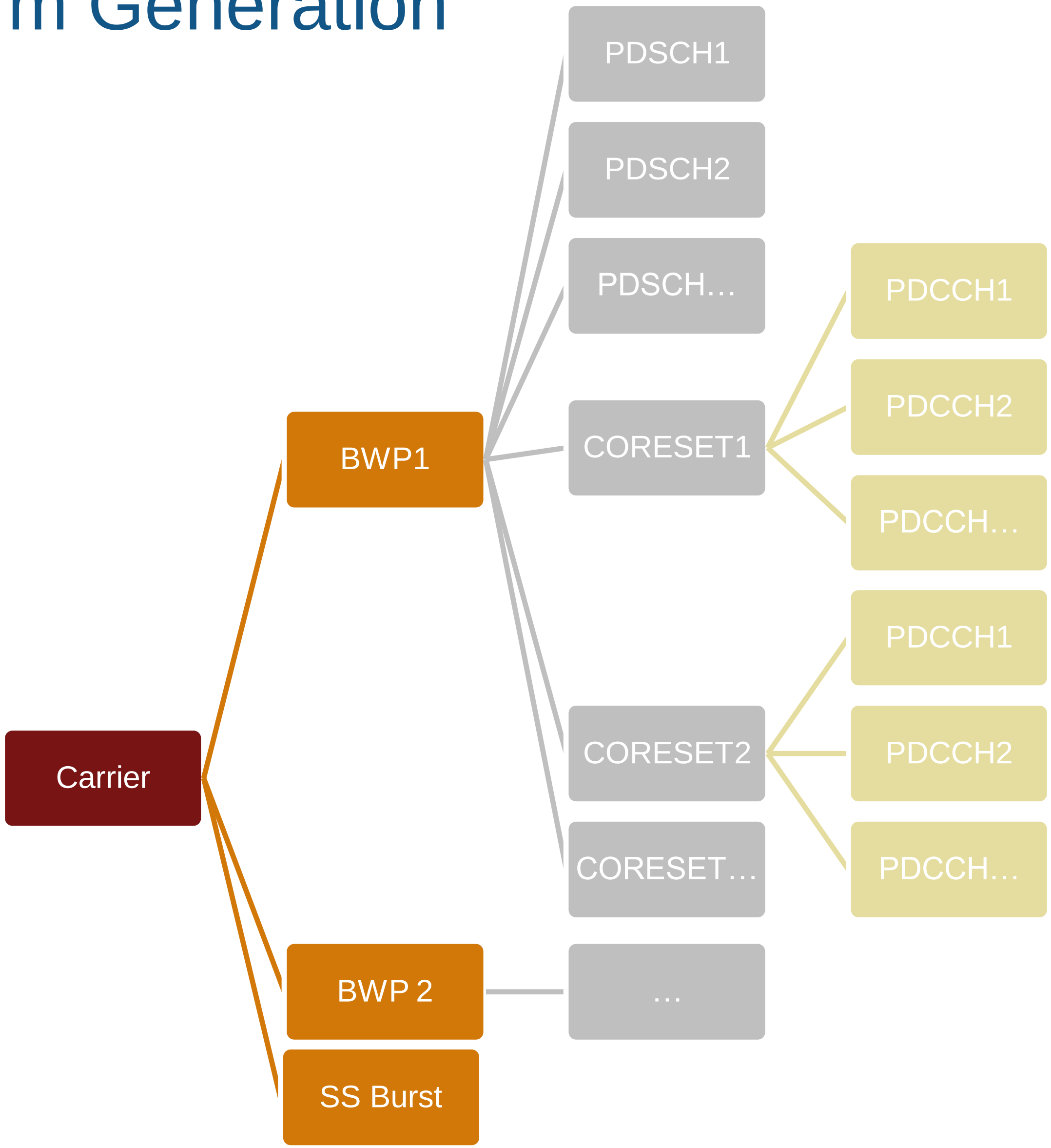


Bandwidth Parts (BWP)

- Carrier bandwidth divided into BWPs
- A BWP is characterised by
  - Subcarrier spacing
  - Cyclic prefix
- Addresses the following issues:
  - Some devices may not be able to receive the full BW
  - Bandwidth adaptation: reduce energy consumption when only narrow bandwidth is required
- 5G Toolbox supports different BWPs



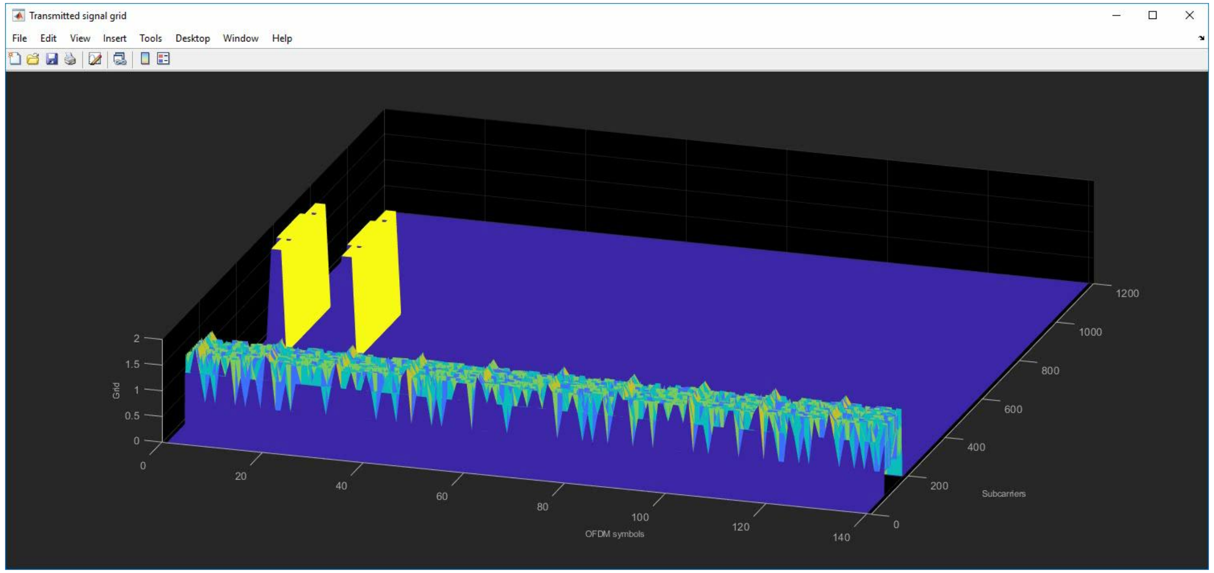
Downlink Waveform Generation



5G waveform generation

- 5G Toolbox supports downlink & uplink waveform generation

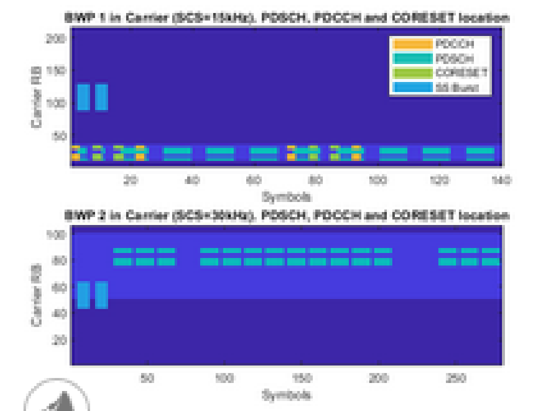
- Generated waveforms feature:
  - mixed numerology
  - multiple bandwidth parts
  - multiple PDSCHs / PUSCHs
  - multiple PDCCHs / PUCCHs
  - fully parameterizable SS bursts
  - multiple CORESETS and search spaces



Power levels have been modified to improve visualization

5G waveform generation in action

Demo

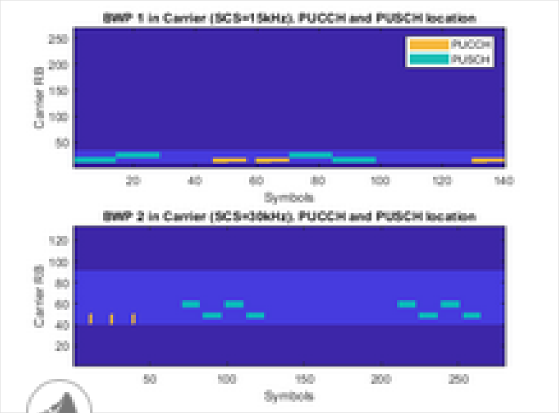


5G NR Downlink Carrier Waveform Generation

Create DL waveforms containing SS burst, PDSCH, PDCCH and DM-RS, with mixed SCS and multiple BWPs.

Open Script

MATLAB Example



5G NR Uplink Carrier Waveform Generation

Create UL waveforms containing PUSCH, PUCCH and DM-RS, with mixed SCS and multiple BWPs.

Open Script

MATLAB Example

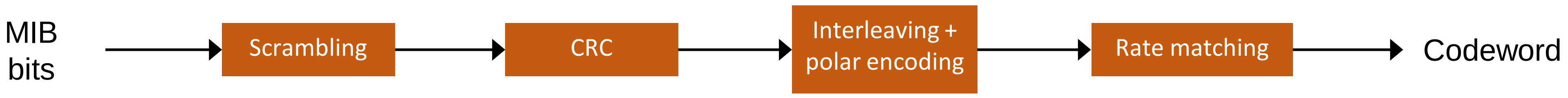
Transport Channels

- Offer information transport services to MAC layer
- Carry control/signalling and data
- Define the scrambling, channel coding, interleaving and rate matching to apply to the information

DL Transport Channels	UL Transport Channels
DL-SCH DL shared channel	**UL-SCH** UL shared channel
DCI Downlink control information	**UCI** Uplink control information
BCH Broadcast channel	**RACH** Random access channel
PCH Paging channel	

Transport Channel Processing: Encoding & Decoding

- BCH



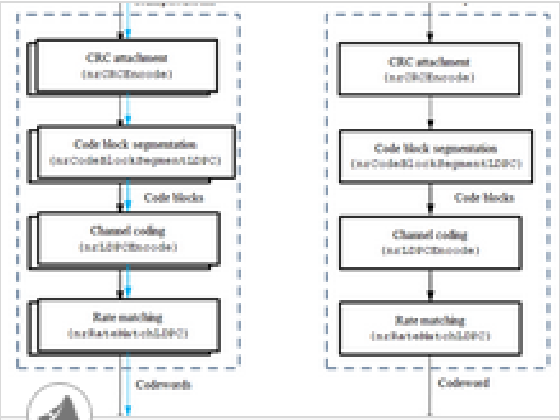
- DL-SCH

- DCI

5G coding for user channels & control channels

Demo

DL-SCH/UL-SCH



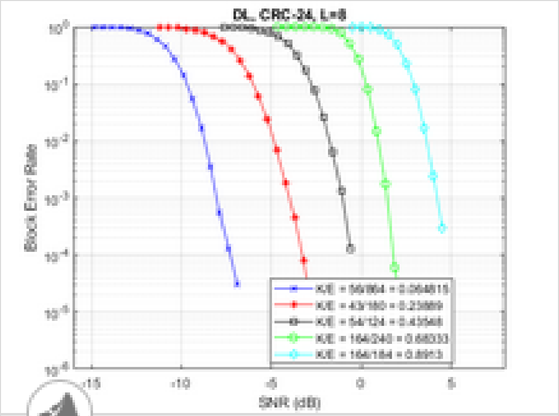
LDPC Processing for DL-SCH and UL-SCH

Process 5G NR DL-SCH and UL-SCH transport blocks using LDPC coding.

Open Script

MATLAB Example

DCI/UCI/BCH



5G New Radio Polar Coding

Model the new CA-Polar coding technique in 5G NR communications system.

Open Script

MATLAB Example

5G Toolbox – PHY Layer Functions

All functions are provided as open MATLAB code

NR Processing Subsystems

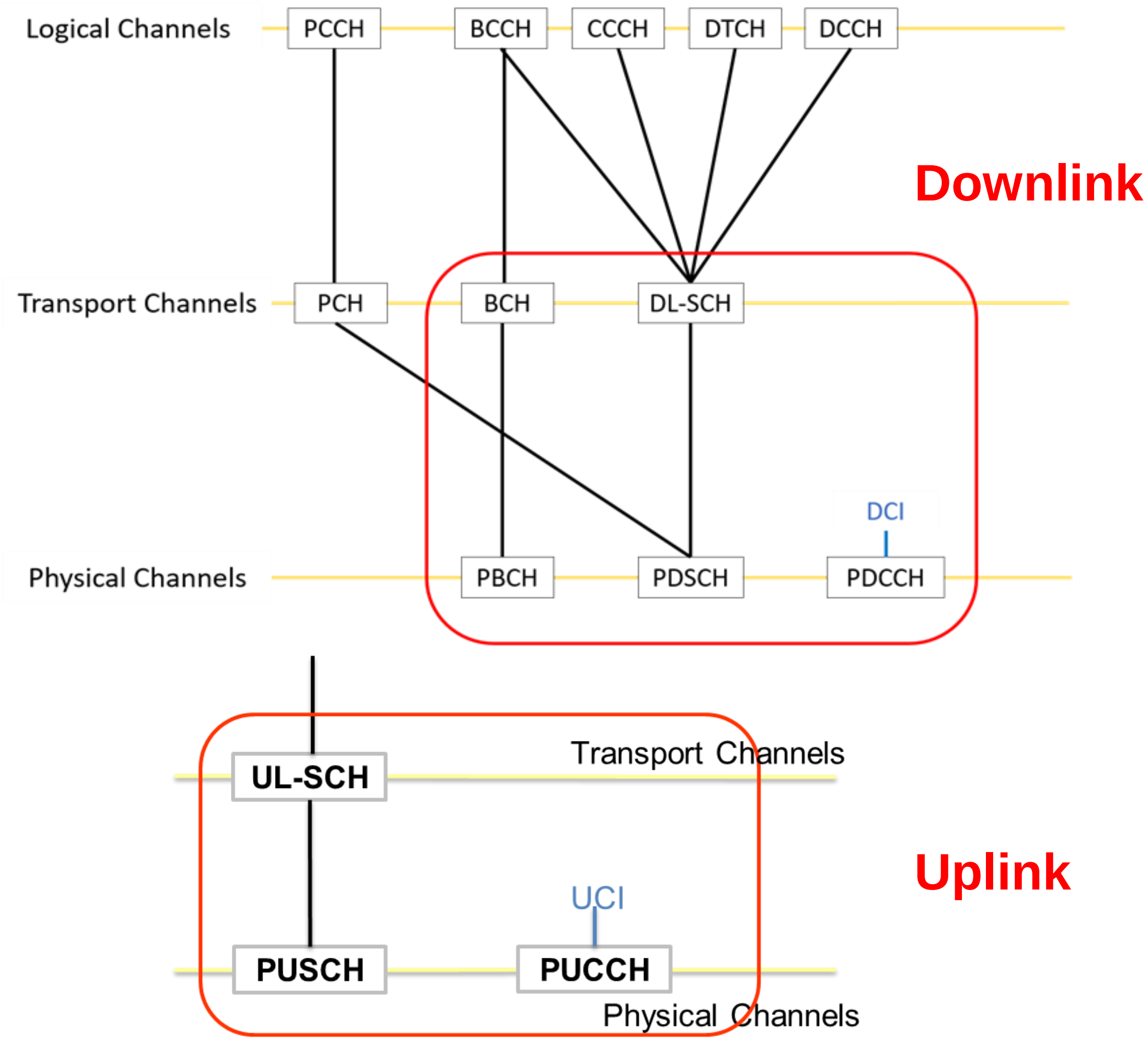
- LPDC & polar coding
- CRC, segmentation, rate matching
- Scrambling, modulation, precoding

NR Downlink and Uplink Channels and Physical Signals

- Synch & broadcast signals
- DL-SCH & PDSCH channels
- DCI & PDCCH channels
- UCI, PUSCH, and PUCCH channels

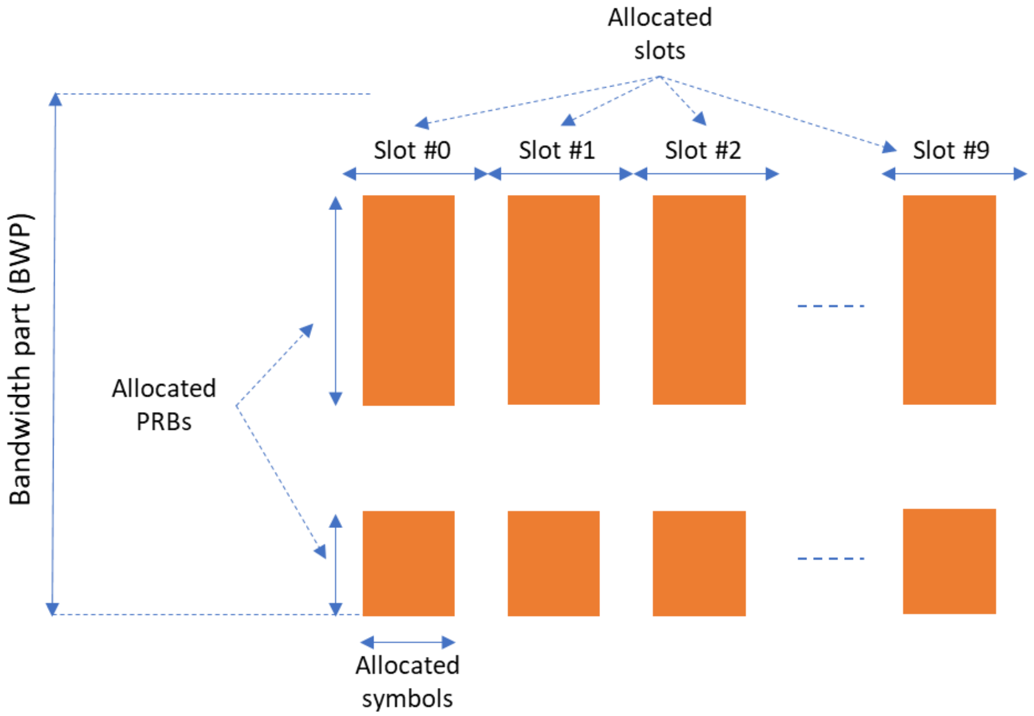
MIMO Prop channels

- TDL & CDL channel models



Fully Configurable PDSCH

- Allocated symbols in the slot
- Slots used for PDSCH
- Period of the allocation (in slots)
- Allocation (PRBs) in the BWP
- Support for multiple PDSCHs




```
pdsch = [];
pdsch(1).Enable = 1;
pdsch(1).BWP = 1;
pdsch(1).Power = 0;
pdsch(1).DataSource = 'PN9';
pdsch(1).TargetCodeRate = 0.4785;
pdsch(1).Xoh_PDSCH = 0;
pdsch(1).Modulation = 'QPSK';
pdsch(1).NLayers = 2;
pdsch(1).RVSequence = [0,1,2,3];
```

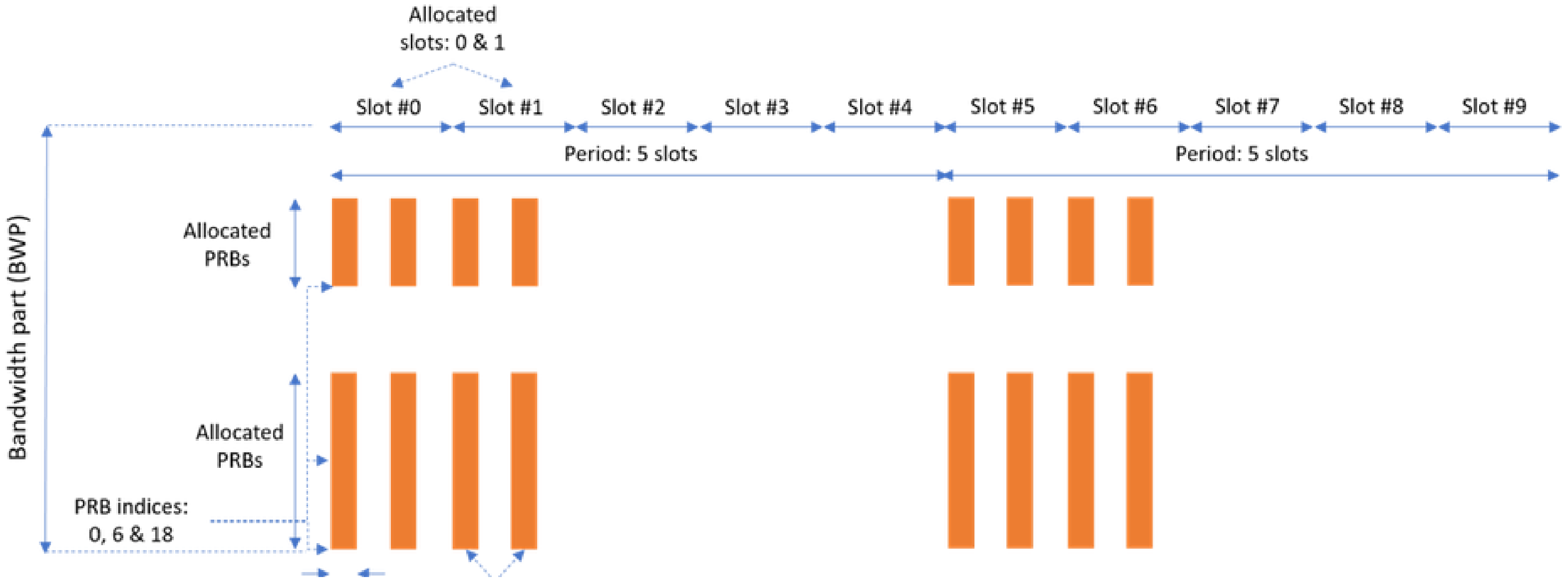
```
-RS configuration (TS 38.211 section
h(1).PortSet = 0:pdsch(1).NLayers-1;
h(1).PDSCHMappingType = 'A';          %
h(1).DL_DMRS_typeA_pos = 2;           %
h(1).DL_DMRS_max_len = 1;             %
h(1).DL_DMRS_add_pos = 0;             %
h(1).DL_DMRS_config_type = 2;         %
pdsch(1).NIDNSCID = 1;                %
pdsch(1).NSCID = 0;                   %
pdsch(1).PowerDMRS = 0;               %
```

```
pdsch(1).AllocatedSymbols = [2:10];     % Range of symbols in a slot
pdsch(1).AllocatedSlots = [0:9];        % Allocated slots indices
pdsch(1).AllocatedPeriod = 15;          % Allocation period in slots
pdsch(1).AllocatedPRB = [0:5, 10:20];   % PRB allocation
pdsch(1).RNTI = 0;                      % RNTI
pdsch(1).NID = 1;                       % Scrambling for data part
```

CORESET(Control Resource Sets) and Search Space Configuration

- 5G Toolbox allows you to specify
  - Possible locations (in time and frequency) of the control channel for a given numerology
  - Allocated OFDM symbols in a slot
  - CORESET duration in symbols, either 1, 2 or 3.
  - The allocated slots
  - Periodicity of the allocation

```
% CORESET/search configurations
coreset = [];
coreset(1).AllocatedSymbols = [0,7];
coreset(1).AllocatedSlots = [0,1];
coreset(1).AllocatedPeriod = 5;
coreset(1).Duration = 3;
coreset(1).AllocatedPRB = 6*[0,1,3];
```



PDCCH Mapping to CORESET: CORESET specification

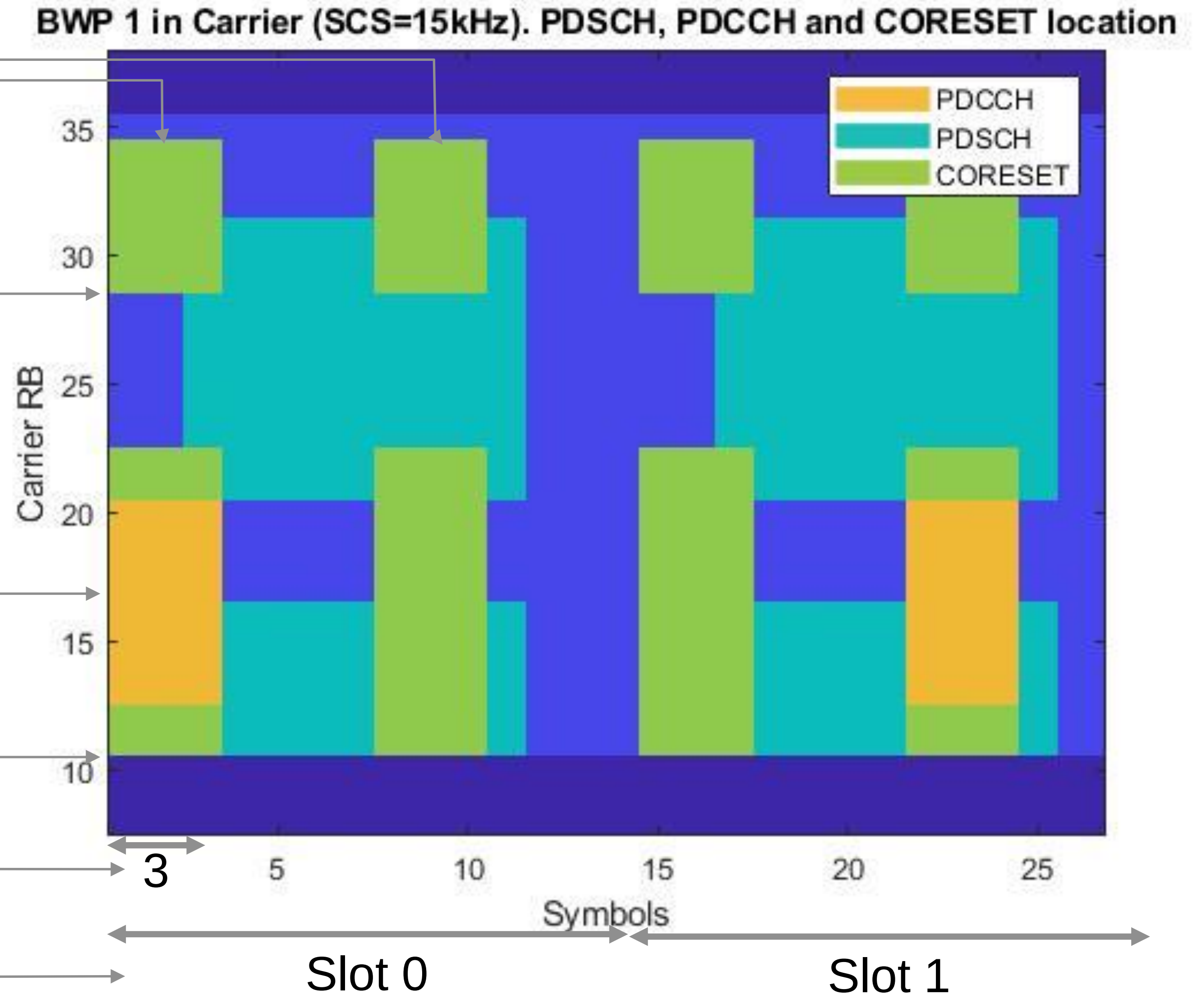
Allocated symbols: 0, 7

```
% CORESET/search configurations
coreset = [];
coreset(1).AllocatedSymbols = [0,7];
coreset(1).AllocatedSlots = [0,1];
coreset(1).AllocatedPeriod = 5;
coreset(1).Duration = 3;
coreset(1).AllocatedPRB = 6*[0,1,3];
```

Allocated PRB: 18
6
0

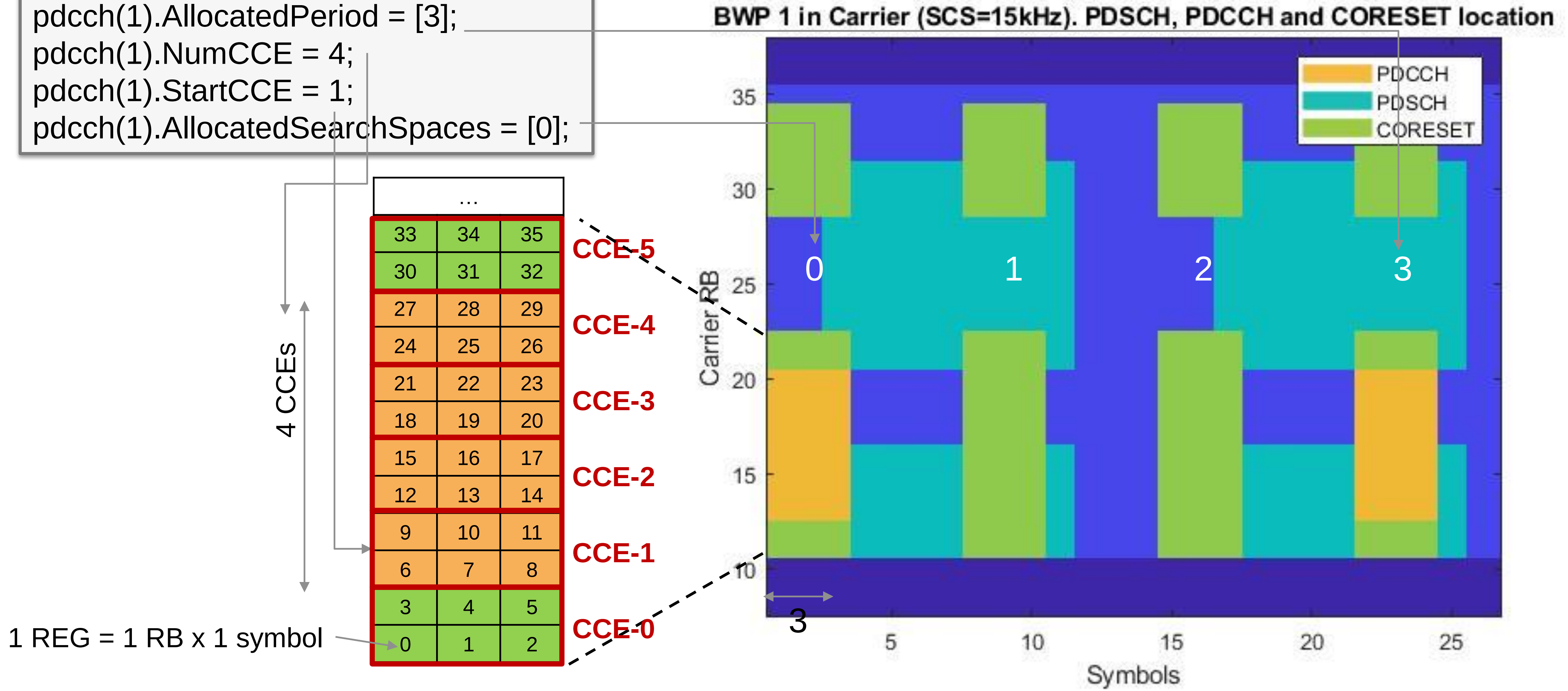
Duration: 3 symbols

Allocated slots: 0, 1



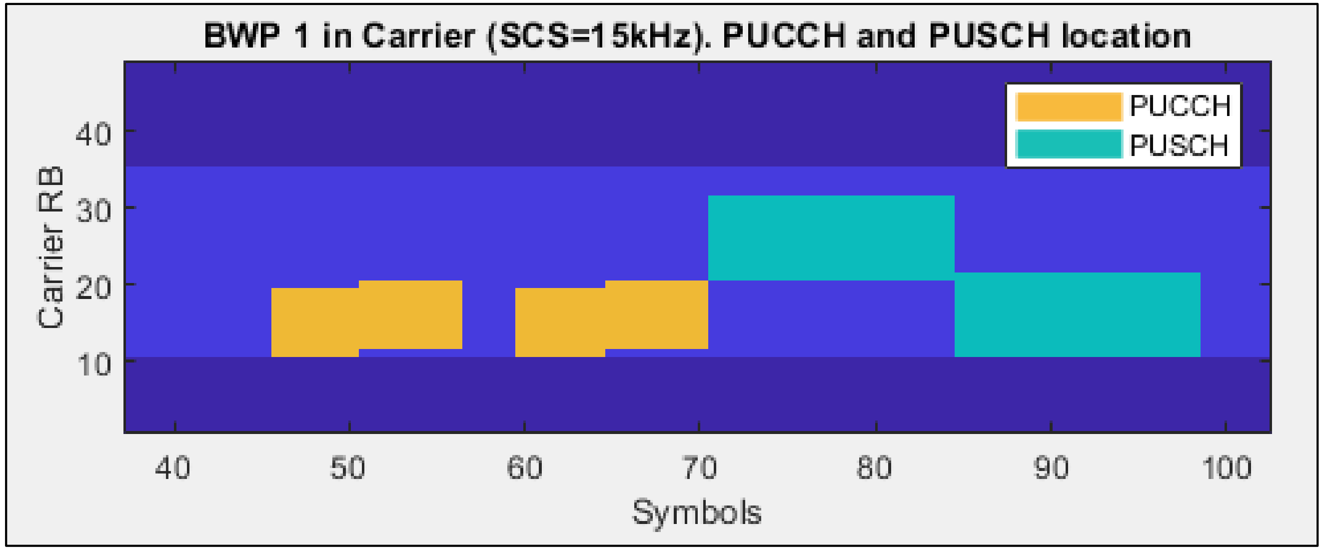
PDCCH Mapping to CORESET: Example

```
pdcch(1).CORESET = 1;
pdcch(1).AllocatedPeriod = [3];
pdcch(1).NumCCE = 4;
pdcch(1).StartCCE = 1;
pdcch(1).AllocatedSearchSpaces = [0];
```



Uplink: Fully Configurable PUSCH

- Allocated symbols in the slot
- Slots used for PUSCH
- Period of the allocation (in slots)
- Allocation (PRBs) in the BWP




```
pusch = [];
pusch(1).Enable = 1;
pusch(1).BWP = 1;
pusch(1).Power = 0;
pusch(1).EnableCoding = 1;
pusch(1).NID = 1;
pusch(1).RNTI = 0;
pusch(1).TransformPrecoding = 0;
pusch(1).TargetCodeRate = 0.47;
pusch(1).Xoh_PUSCH = 0;
```

```
% Transmission settings
pusch(1).TxScheme = 'codebook';
pusch(1).Modulation = 'QPSK';
pusch(1).NLayers = 2;
pusch(1).NAntennaPorts = 4;
pusch(1).TPMI = 0;
pusch(1).RVSequence = [0 2 3 1];
pusch(1).IntraSlotFreqHopping = 'disabled';
pusch(1).RBOffset = 10;

% Multi-slot transmission
pusch(1).InterSlotFreqHopping = 'enabled';
```

```
pusch(1).PUSCHMappingType = 'A';        % PUSCH mapping type ('A'(slot-wise),'B'(non slot-wise))
pusch(1).AllocatedSymbols = 0:13;       % Range of symbols in a slot
pusch(1).AllocatedSlots = [0 1];        % Allocated slots indices
pusch(1).AllocatedPeriod = 5;           % Allocation period in slots (empty implies no repetition)
pusch(1).AllocatedPRB = 0:10;           % PRB allocation
```

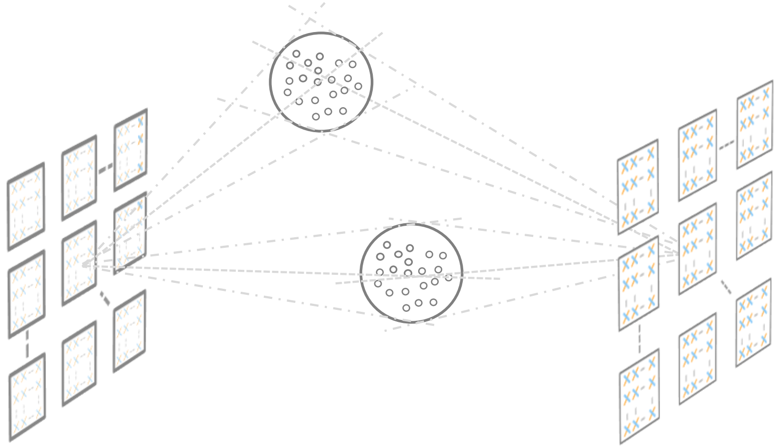
Uplink: PUCCH Configuration

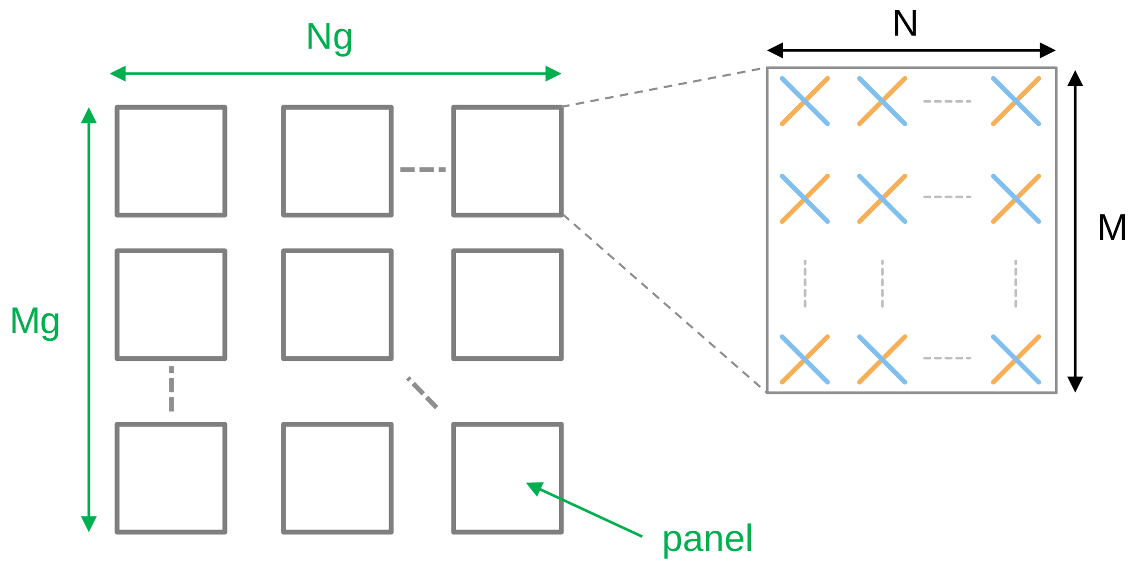
- Fully configurable set of parameters
- Support for multiple PUCCH instances

```
% Format specific parameters
pucch(1).PUCCHFormat = 3;         % PUCCH format 0/1/2/3/4
pucch(1).StartSymbol = 3;         % Starting symbol index
pucch(1).NrOfSymbols = 11;        % Number of OFDM symbols allocated for PUCCH
pucch(1).InitialCS = 3;           % Initial cyclic shift for format 0 and 1
pucch(1).OCCI = 0;                % Orthogonal cover code index for format 1 and 4
pucch(1).Modulation = 'QPSK';     % Modulation for format 3/4 ('pi/2-BPSK','QPSK')
pucch(1).NrOfRB = 9;              % Number of resource blocks for format 2/3
pucch(1).SpreadingFactor = 4;     % Spreading factor for format 4, value is either 2 or 4
pucch(1).AdditionalDMRS = 1;      % Additional DM-RS (0/1) for format 3/4

pucch(1).RNTI = 0;                % RNTI (0...65535) for formats 2/3/4
pucch(1).NID = 1;                 % PUCCH scrambling identity (0...1023) for formats 2/3/4
pucch(1).HoppingId = 1;           % PUCCH hopping identity (0...1023) for formats 0/1/3/4
pucch(1).NIDDMRS = 1;             % DM-RS scrambling identity (0...65535) for PUCCH format 2
```

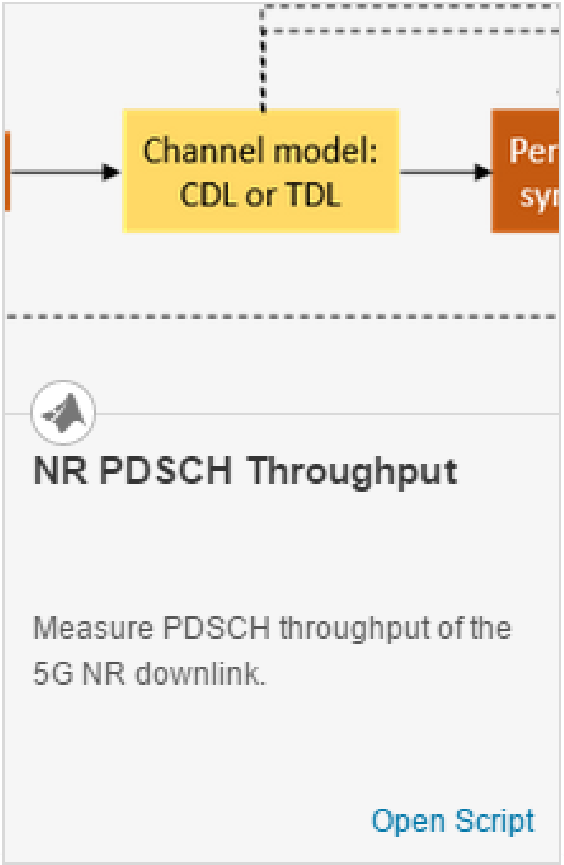
5G Channel Models

- Implementation of 5G channel models TR 38.901
- These include control of:
  - Delay profile: TDL and CDL profiles: A, B, C, D, E or custom
  - Channel delay spread
  - Doppler shift
  - MIMO correlation
  - CDL: spatial channel model, includes also:
    - Antenna array geometry [M, N, P, Mg, Ng]



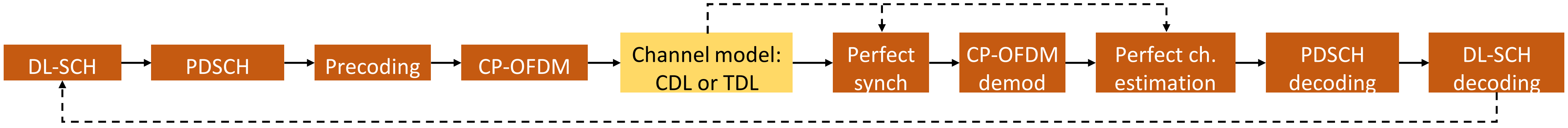


5G downlink end-to-end link-level throughput measurements



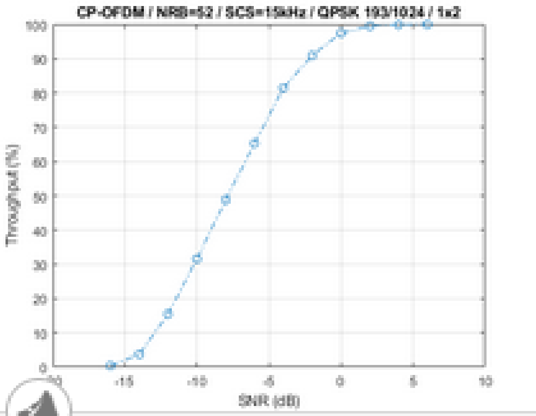
MATLAB Example

Demo



5G uplink end-to-end link-level throughput measurements

Demo

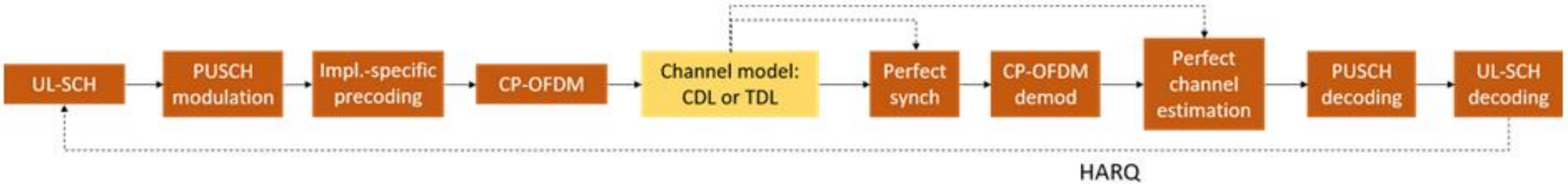


NR PUSCH Throughput

Measure PUSCH throughput of the 5G NR uplink.

Open Script

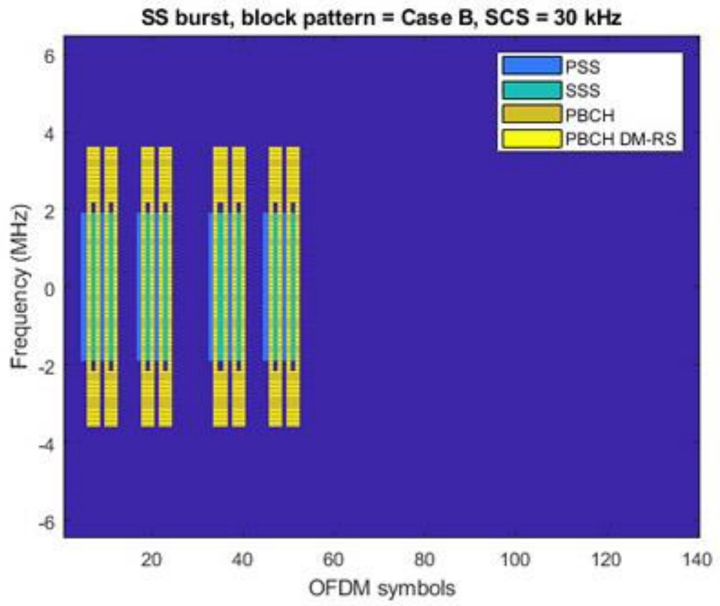
MATLAB Example



Synchronization

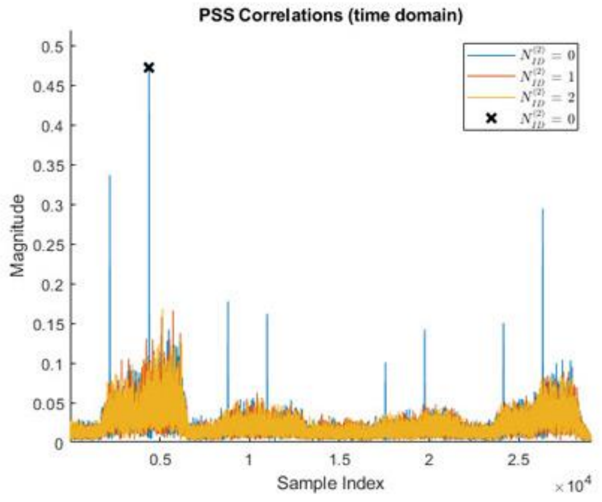
Demo

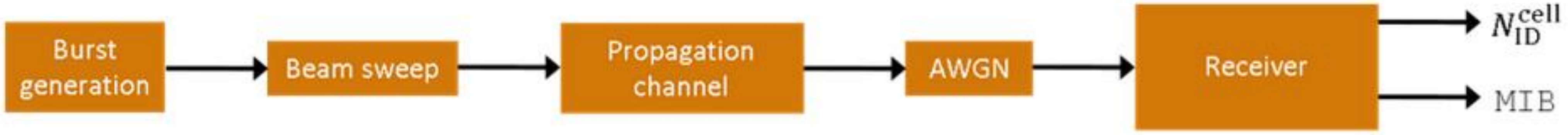
- Construct synchronization signal (SS) bursts
- Pass waveform through fading channel
- Synchronize to receive waveform using
  - Primary synchronization signal (PSS)
  - Secondary synchronization signal (SSS)
  - PBCH demodulation reference signal (PBCH DM-RS)
- Perform PBCH decoding and parsing



Cell search and selection procedures

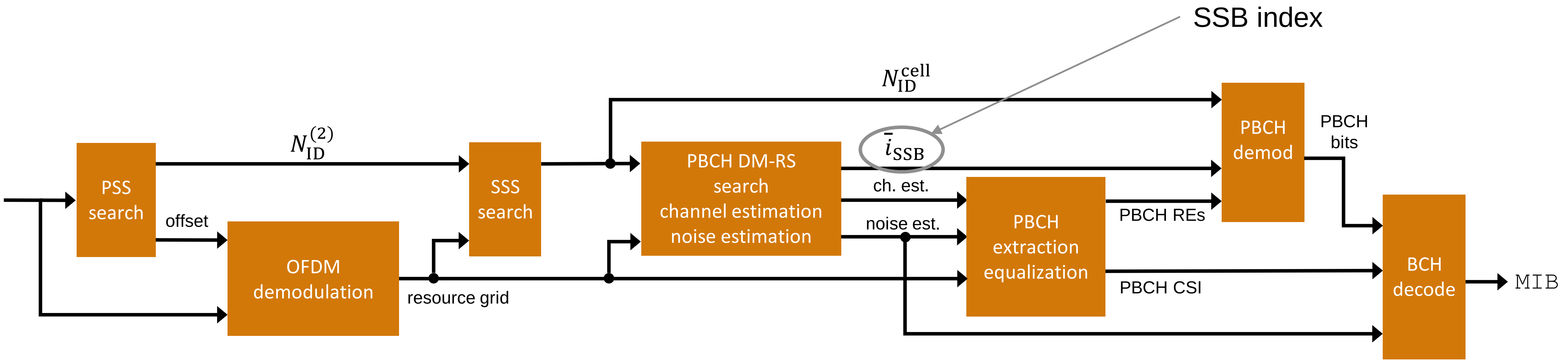
- Obtain cell ID and initial system information including Master Information Block (MIB)
- Perform the following steps:
  - Burst generation
  - Beam sweep
  - TDL propagation channel model and AWGN
  - Receiver synchronization and demodulation





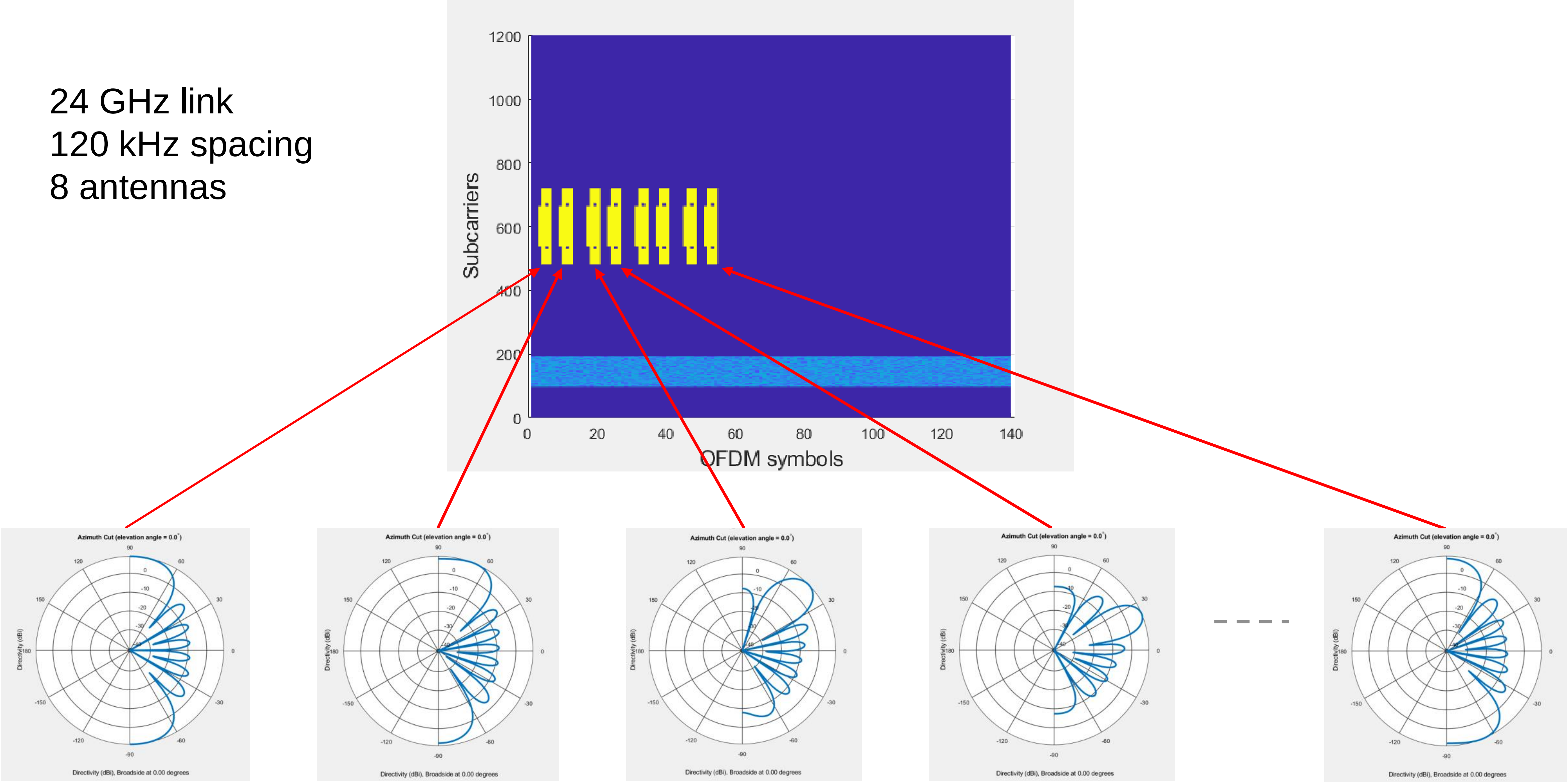
SS Block Decoding (Detailed Diagram)

Demo

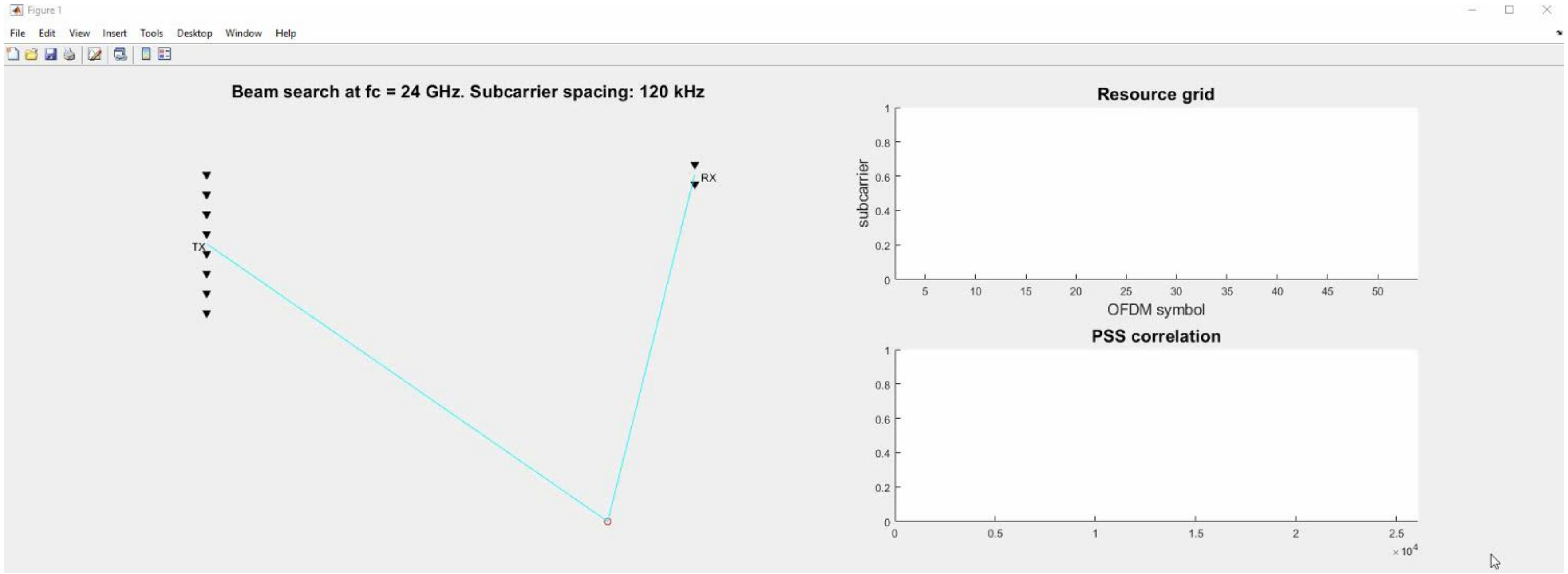


5G mmWave beam search

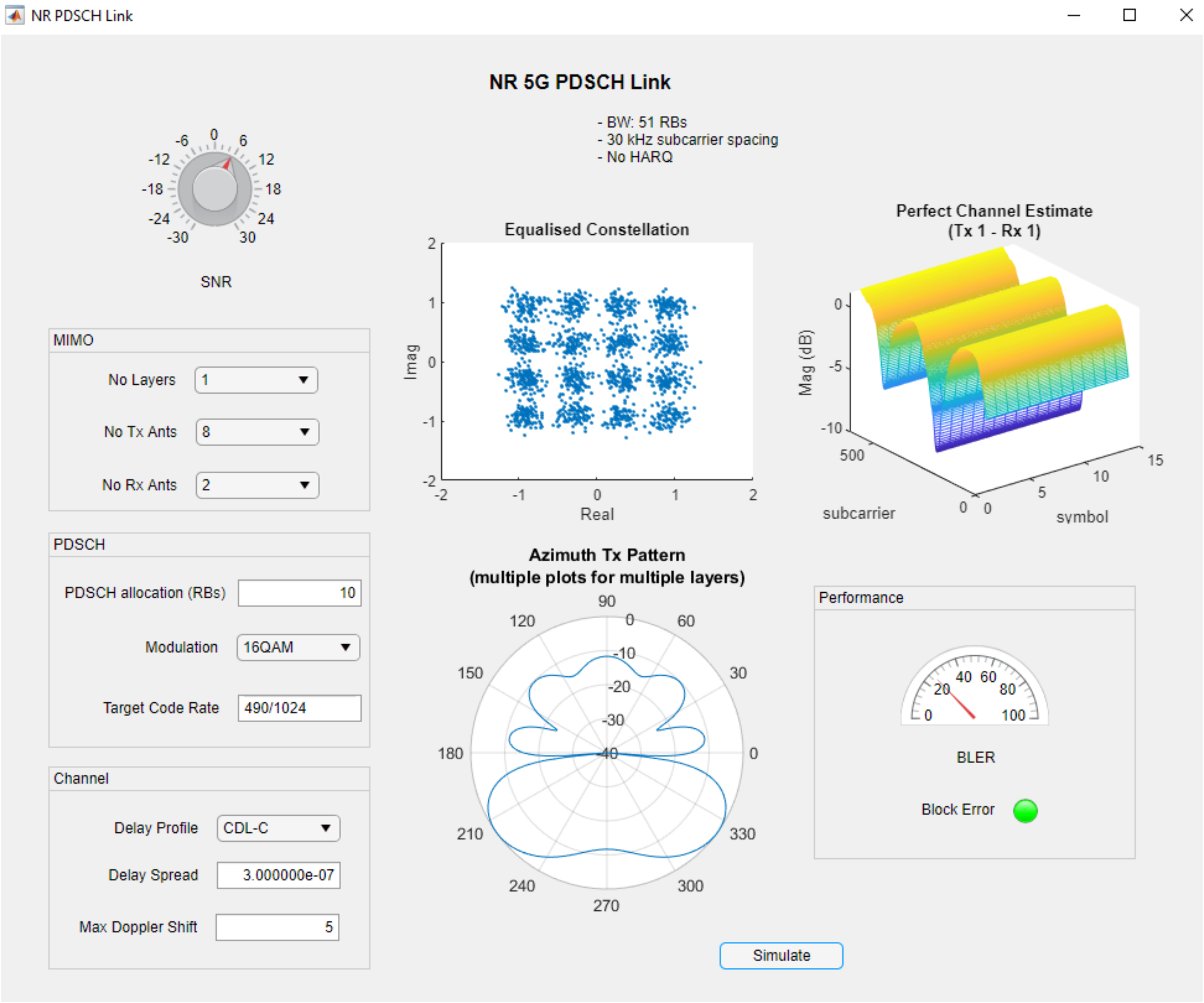
24 GHz link
120 kHz spacing
8 antennas



5G mmWave beam search video



NR 5G Link GUI Demo



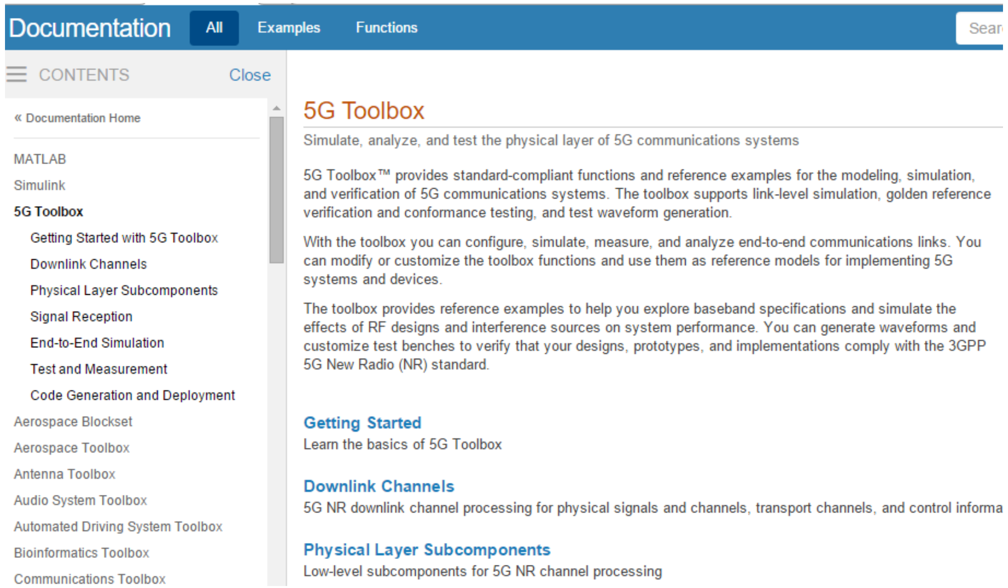
How to learn more

- Go to 5G Toolbox product page
  www.mathworks.com/products/5g

- Watch the 5G Toolbox video
  - On YouTube
  - On 5G Toolbox product page

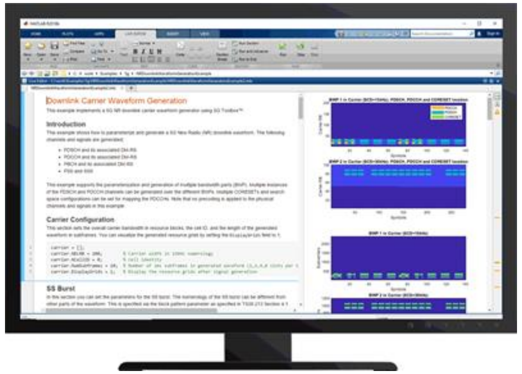
- Go to 5G Toolbox documentation page

- Consult ‘MATLAB for 5G’ page



Summary

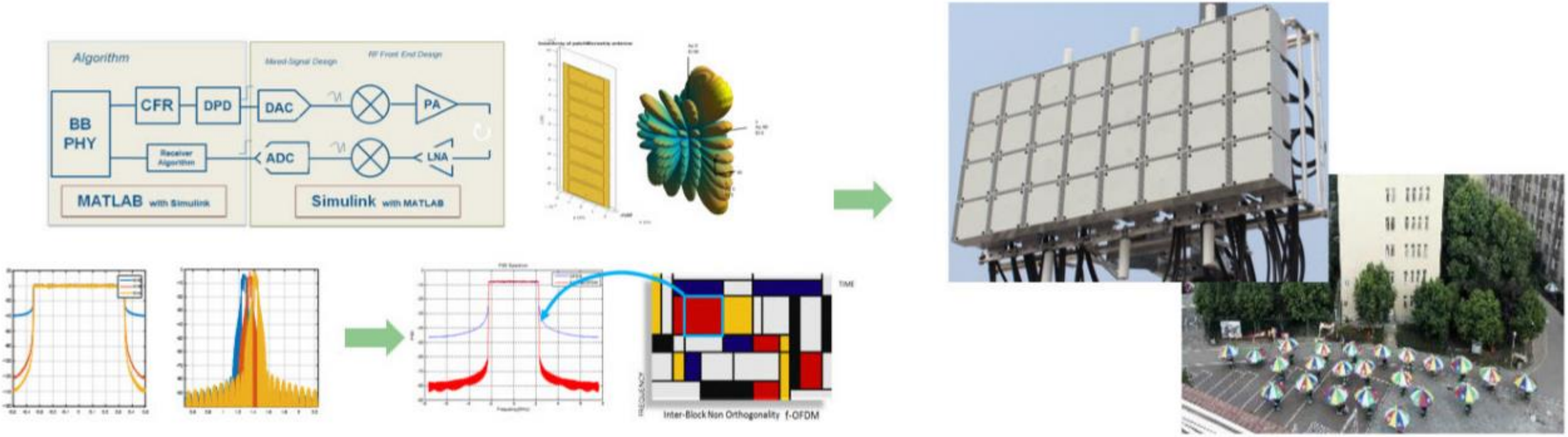
- 5G Toolbox enables you to simulate, analyse, test 5G wireless communications systems
- Standard-compliant MATLAB functions based on Rel. 15 of 3GPP 5G NR standard
  - Waveform generation
  - Downlink & Uplink Physical Channels and Signals including
  - Link-level simulation including PDSCH and PUSCH Throughput Simulation
  - Cell search procedure with MIB decoding
  - LDPC and Polar Coding algorithms
  - TR 38.901 CDL and TDL Propagation Channels



Model Based Design for Implementing 5G NR

5G Prototyping at Huawei

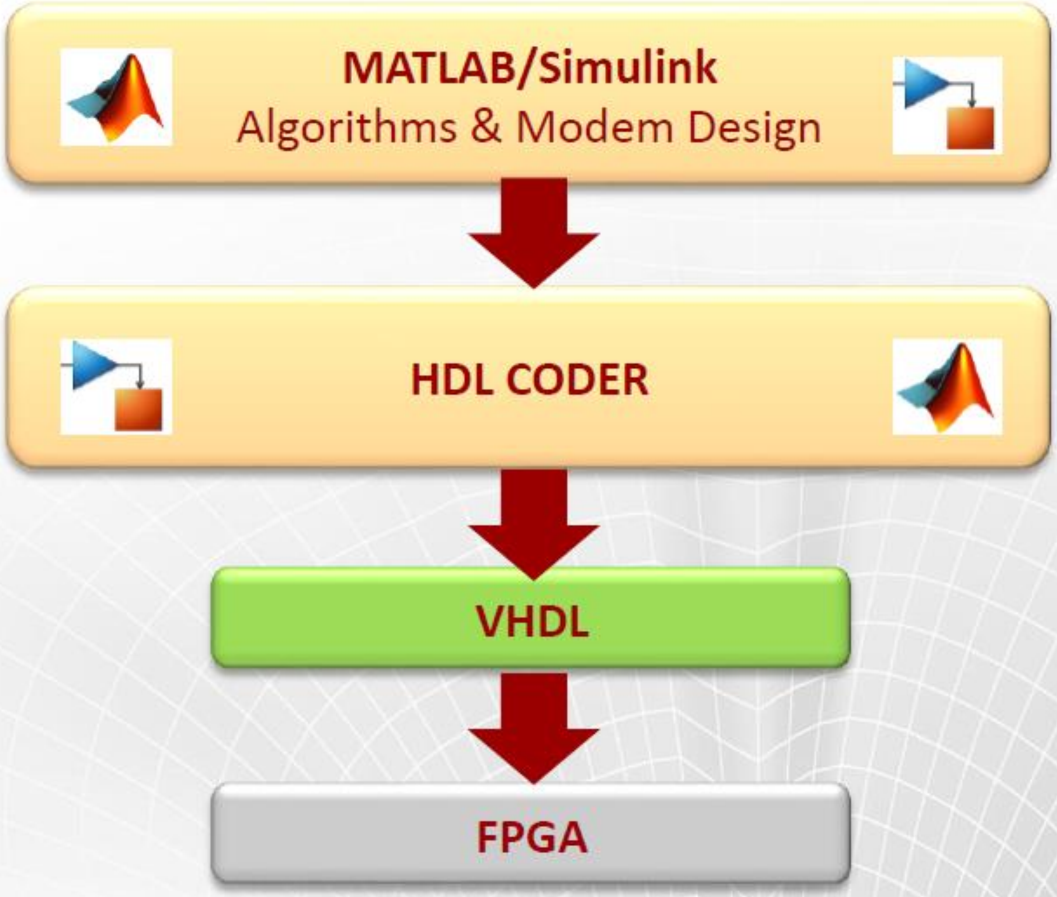
- Leverage existing mature solutions to build the system
- Develop the specific innovation such as new waveforms



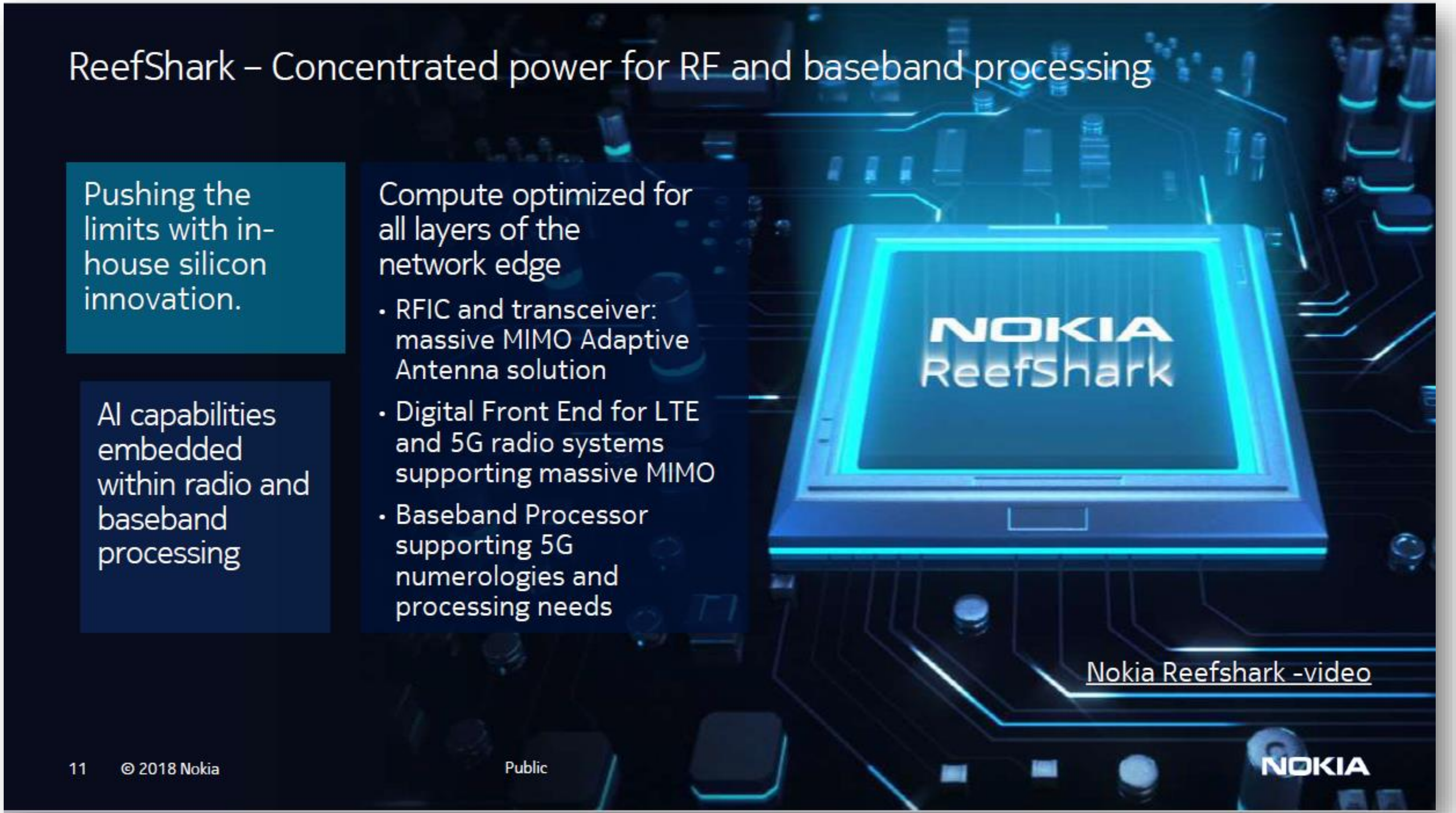
Next generation wireless technology design, verification, and field testing.

Modem: Design and Simulation

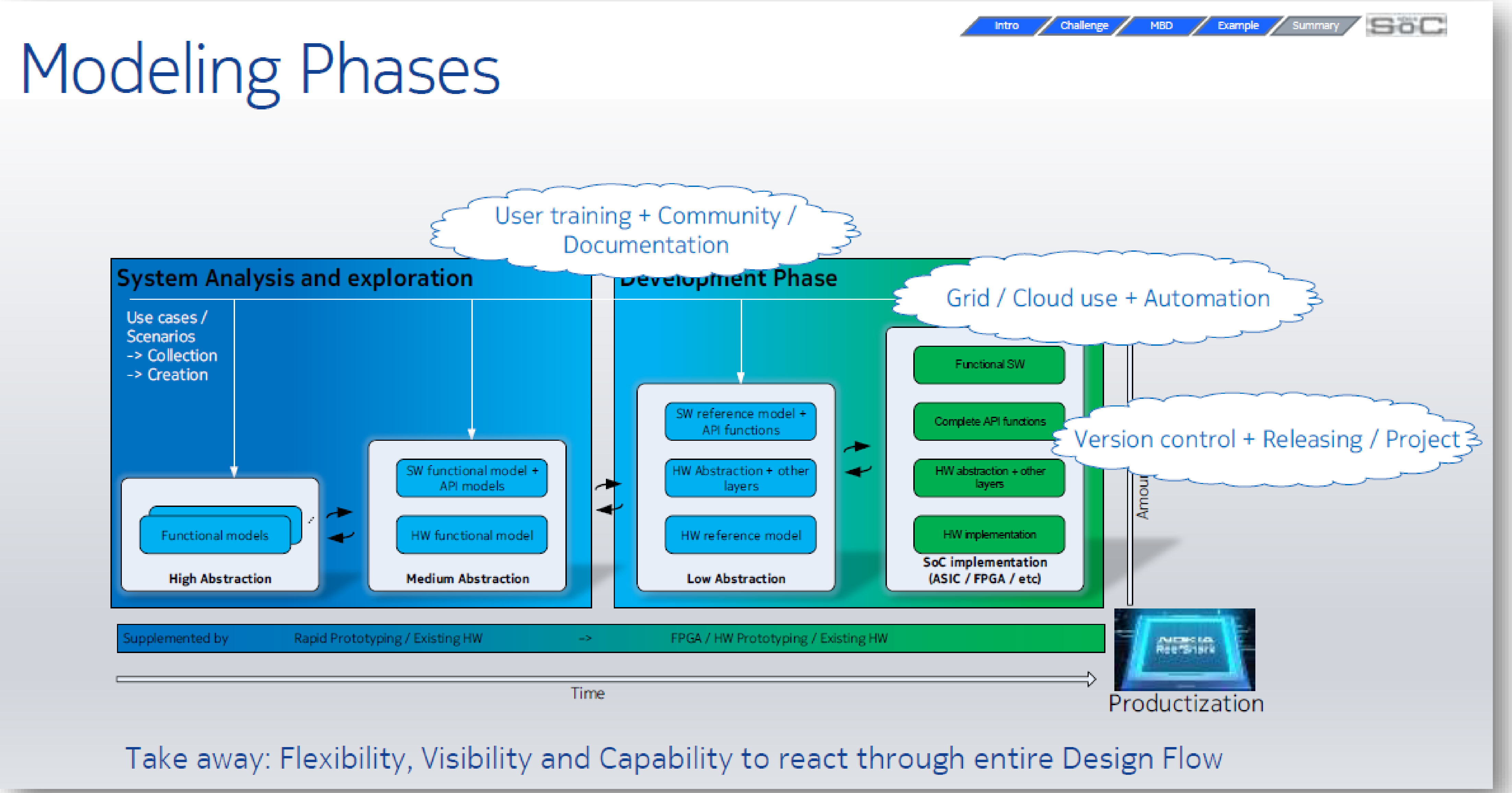
- Design, execute and verify modem in MATLAB/Simulink
- Automatically generation of VHDL code
- Deploy generated code on HW/FPGA



Model Based Design in Nokia 5G

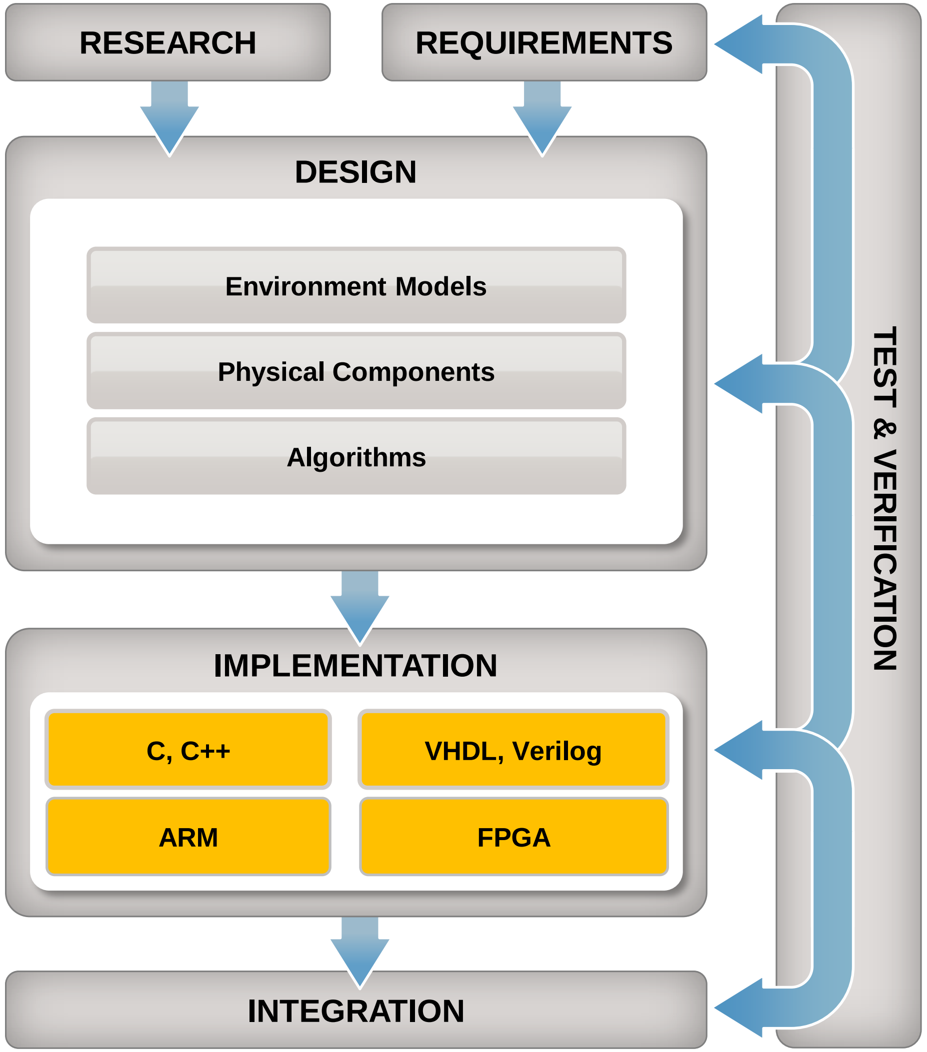


Model Based Design in Nokia 5G



Model-Based Design

From Concept to Production

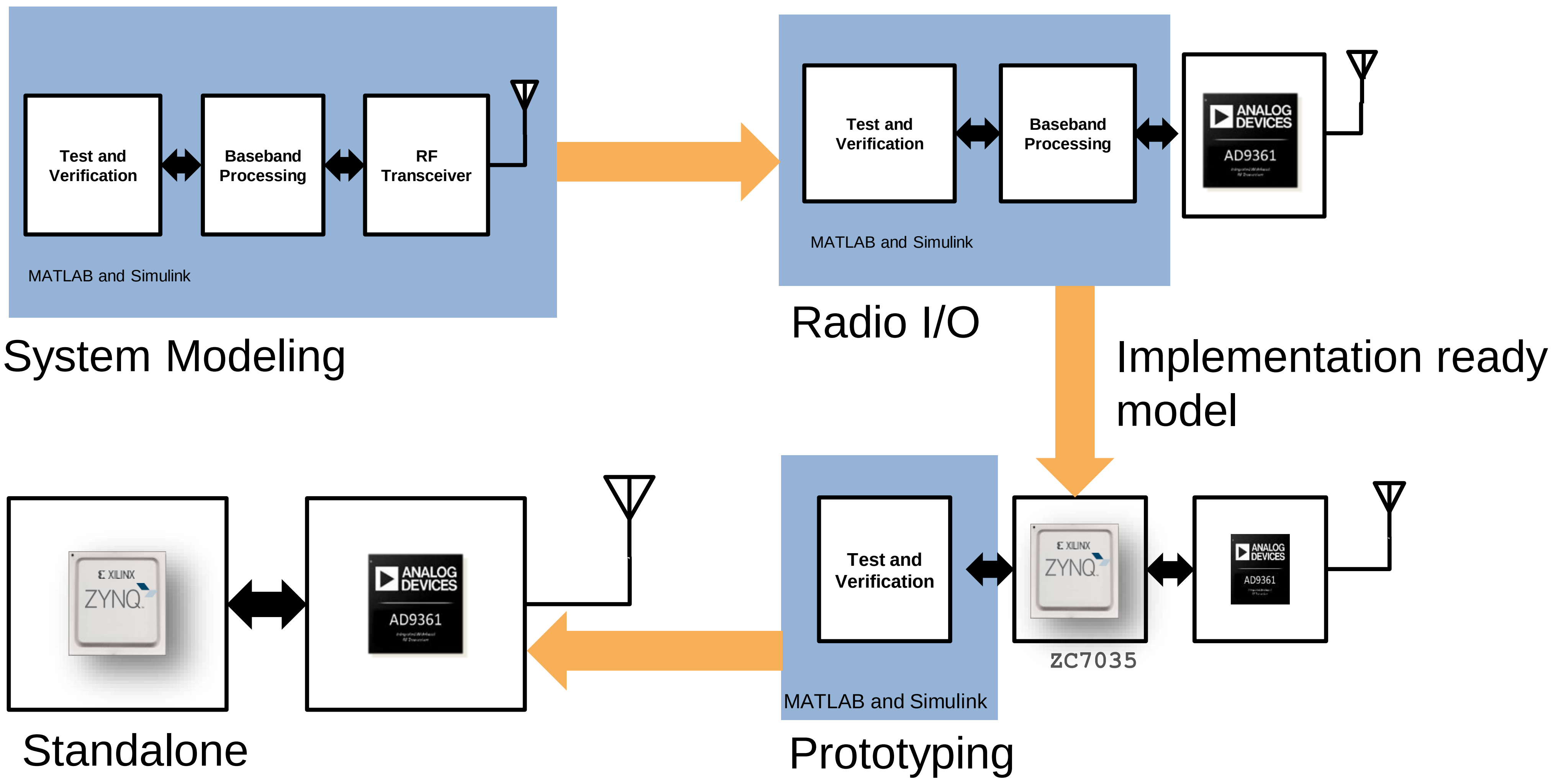


- Model multi-domain systems
- Explore and optimize system behavior in floating point and fixed point
- Collaborate across teams and continents

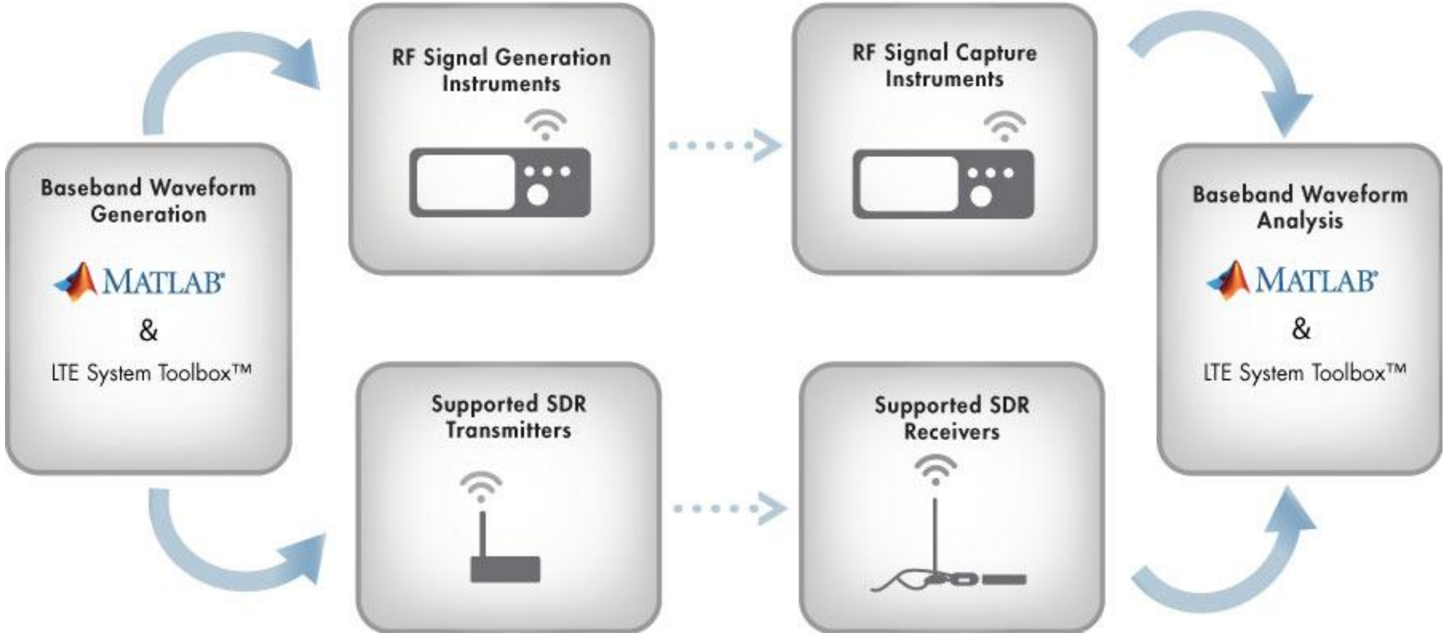
- Generate efficient code
- Explore and optimize implementation tradeoffs

- Automate regression testing
- Detect design errors
- Support certification and standards

Workflow for 5G NR Design and Prototyping



Test with Hardware and Over-the-Air Signals



- Test your algorithms with real signals and scenarios
- Deployment to SDR platforms or your own hardware

Use Supported Hardware…

RF Signal Generator

Spectrum Analyzer

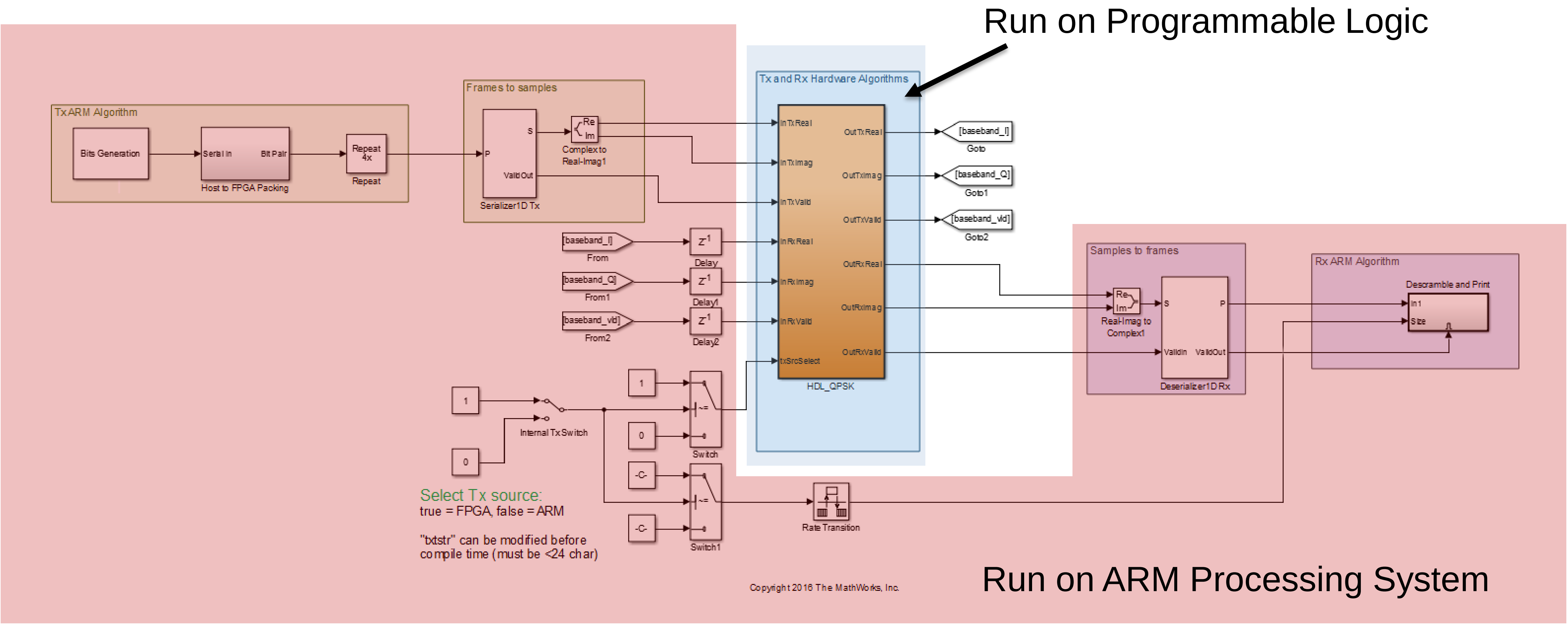
Zynq Radio SDR

USRP SDR

…Or Your Own Hardware

HDL Coder and Embedded Coder to implement your design on FPGA and DSP platforms

Targetable Receiver/Transmitter Model



Key Features in HDL Coder

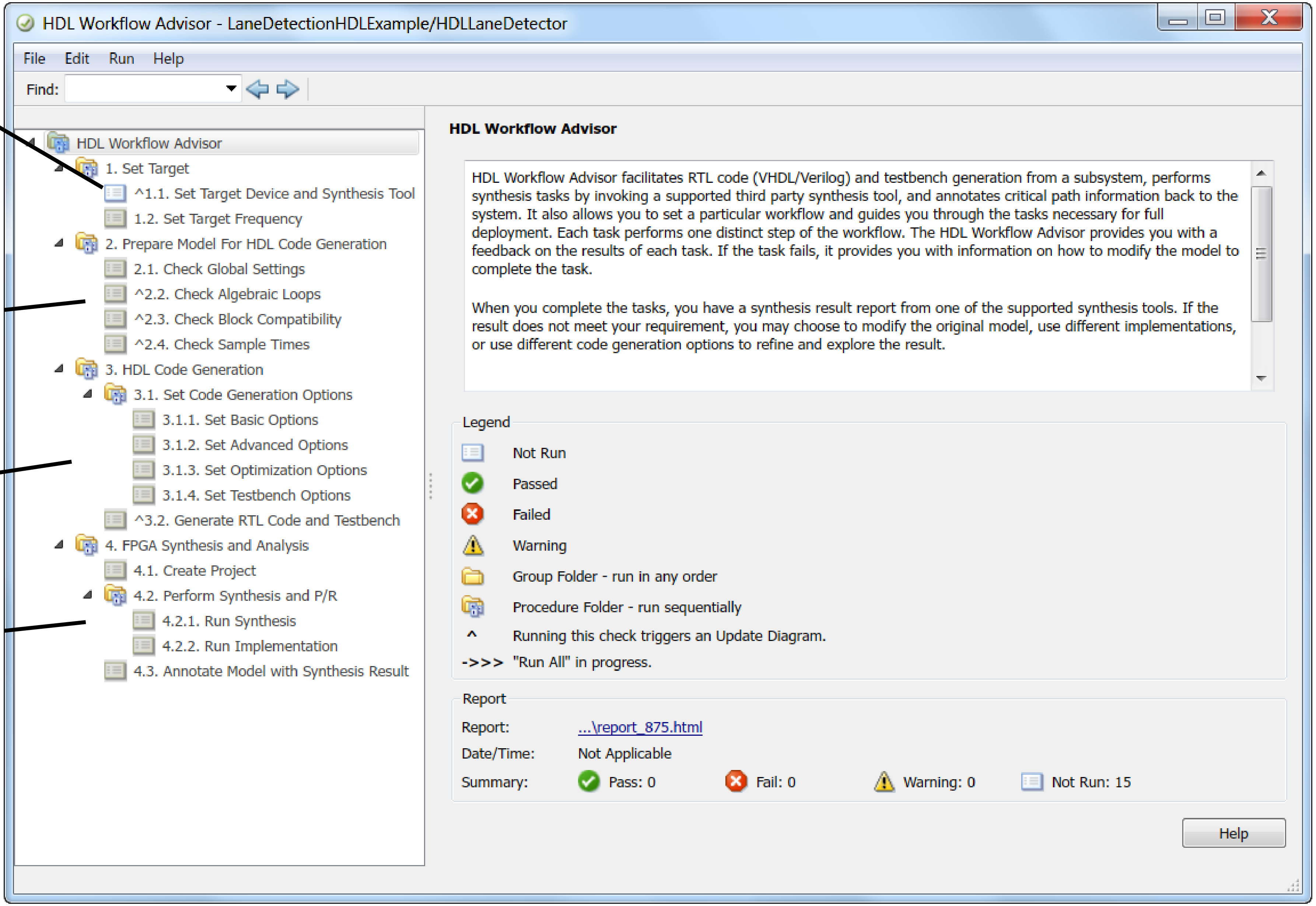
Workflow Advisor Speeds Adoption

- Generic ASIC/FPGA
- IP core generation (AXI)
- FPGA-in-the-loop
- Simulink RealTime I/O

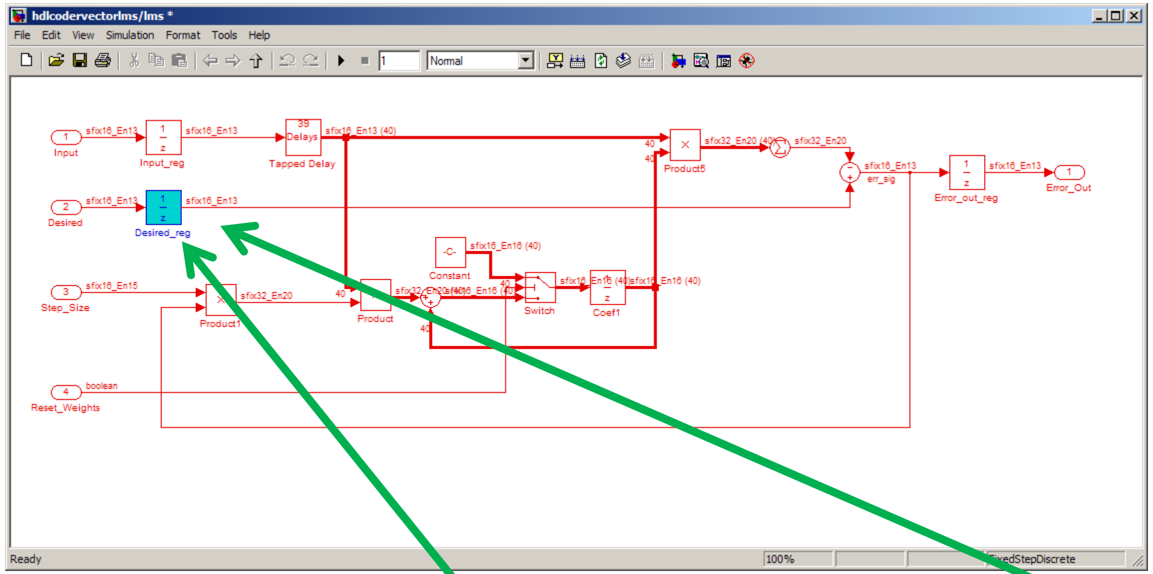
- Check for HDL readiness
- Automatic fixes available

- Optimization settings
- Floating point operations
- Coding style/standards
- Generate testbench

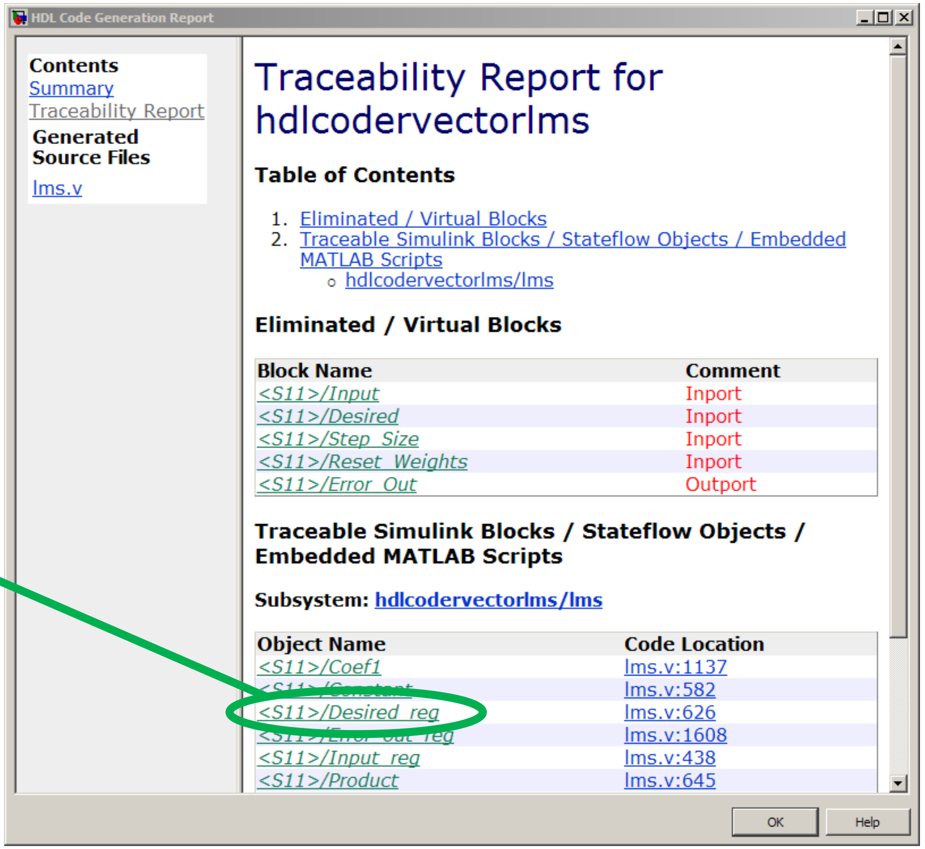
- Run Vivado/Quartus
- Backannotate results



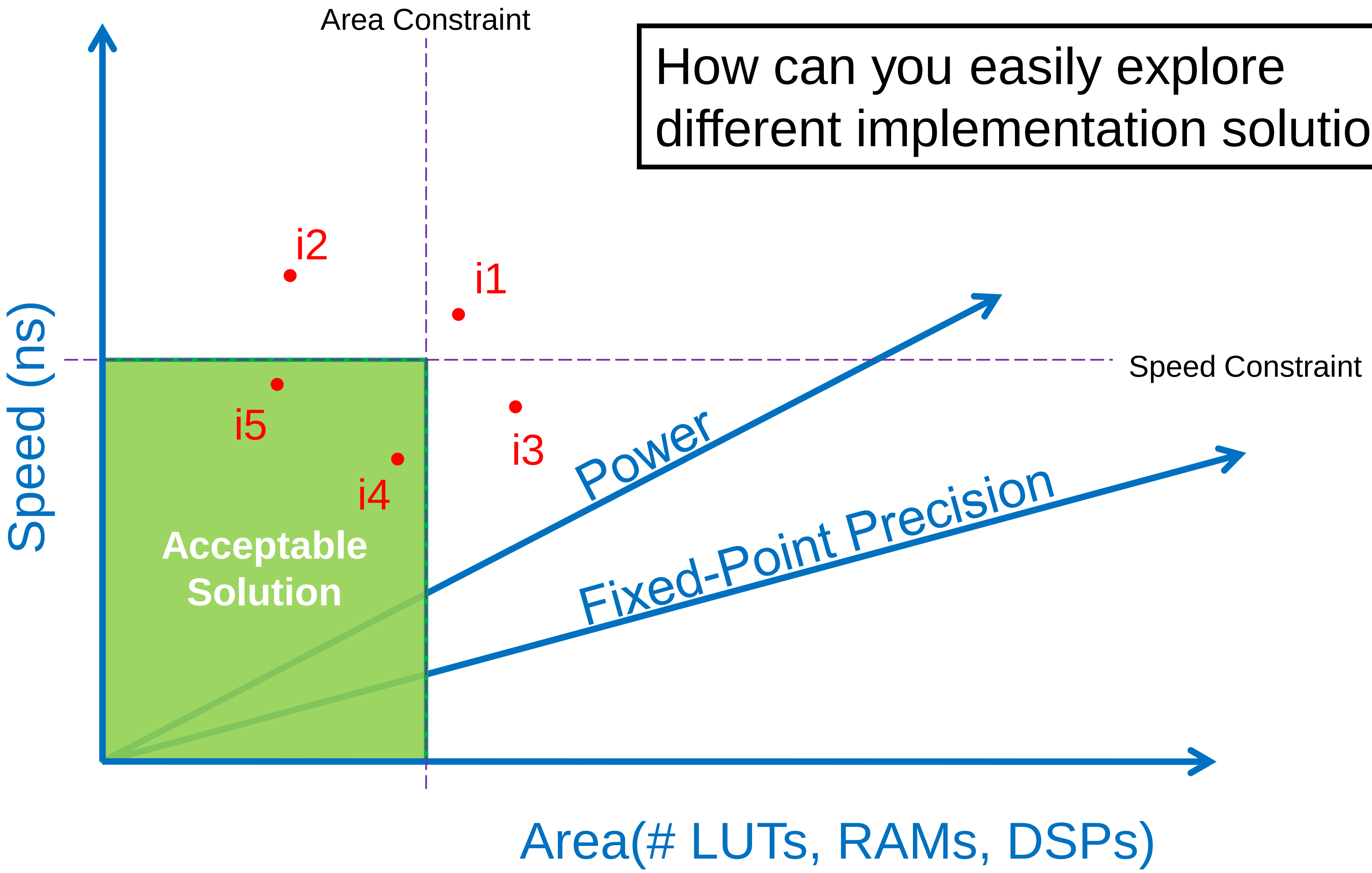
Generate Readable, Traceable Code




```
assign Constant_out1[38] = 0;
assign Constant_out1[39] = 0;



// <S11>/Desired_reg
always @(posedge clk or posedge reset)
  begin : Desired_reg_process
    if (reset)
      Desired_reg_out1 <= 0;
    else
      if (enb)
        Desired_reg_out1 <= Desired;

  end
```



Design Space Exploration



How can you easily explore different implementation solutions?

Easily Identify Timing Bottlenecks
Timing Optimization

Critical Path Report for
my_equalizer_sim_optimization/EqualizerAlgorithm

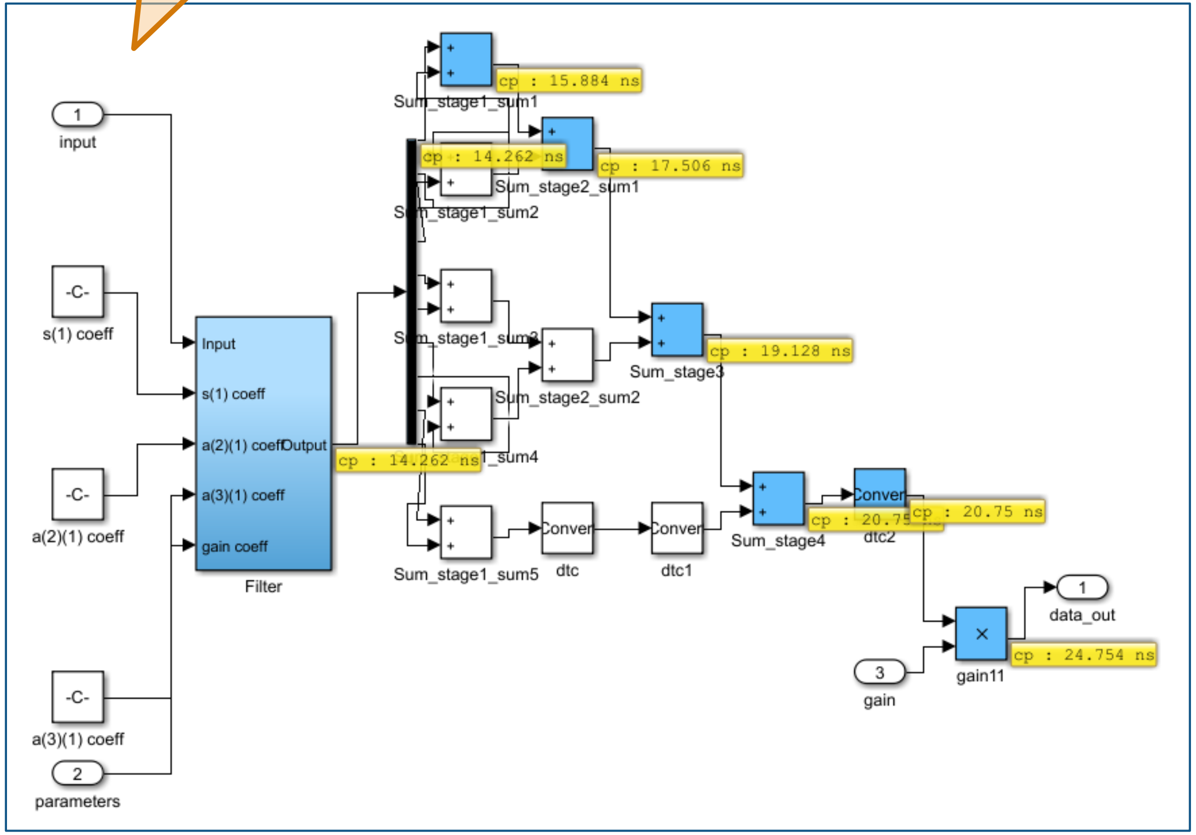
Summary Section

Critical Path Delay : 24.784 ns
Critical Path Begin : Unit Delay
Critical Path End : Unit Delay
Highlight Critical Path: hdl_prj\hdlsrc\my_equalizer_sim_optimization\criticalPathEstimated.m
Highlight Uncharacterized blocks:
hdl_prj\hdlsrc\my_equalizer_sim_optimization\highlightCriticalPathEstimationOffendingBlocks.m

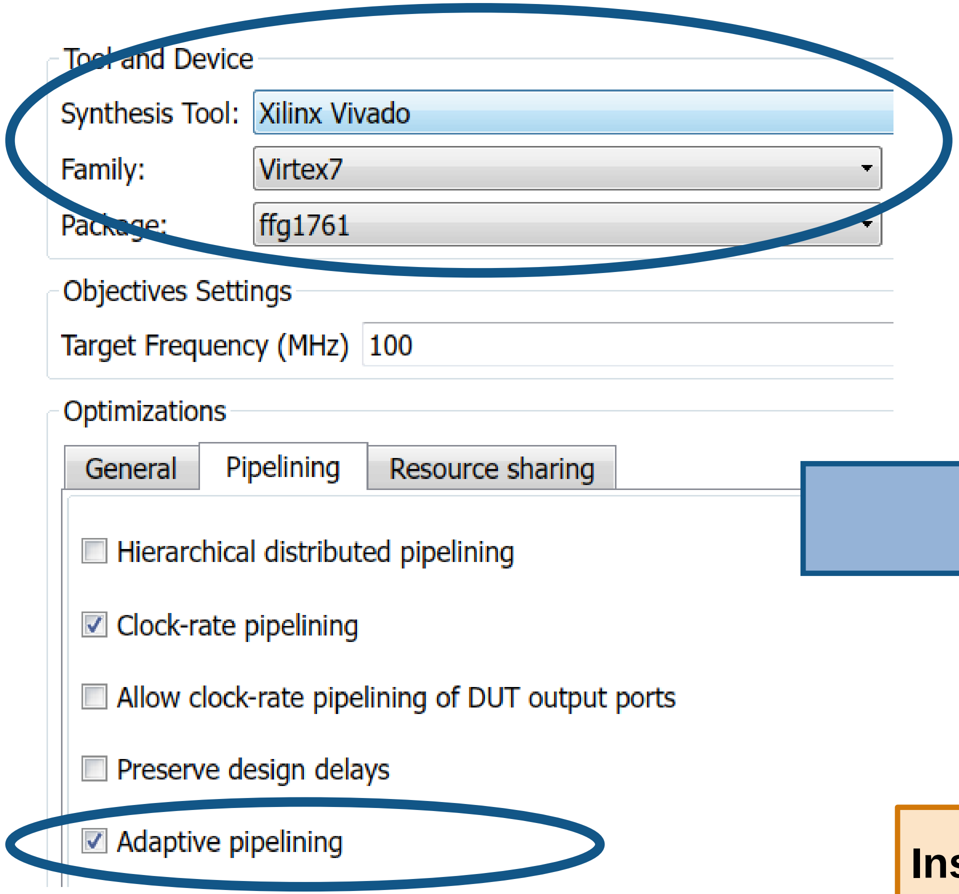
Critical Path Details

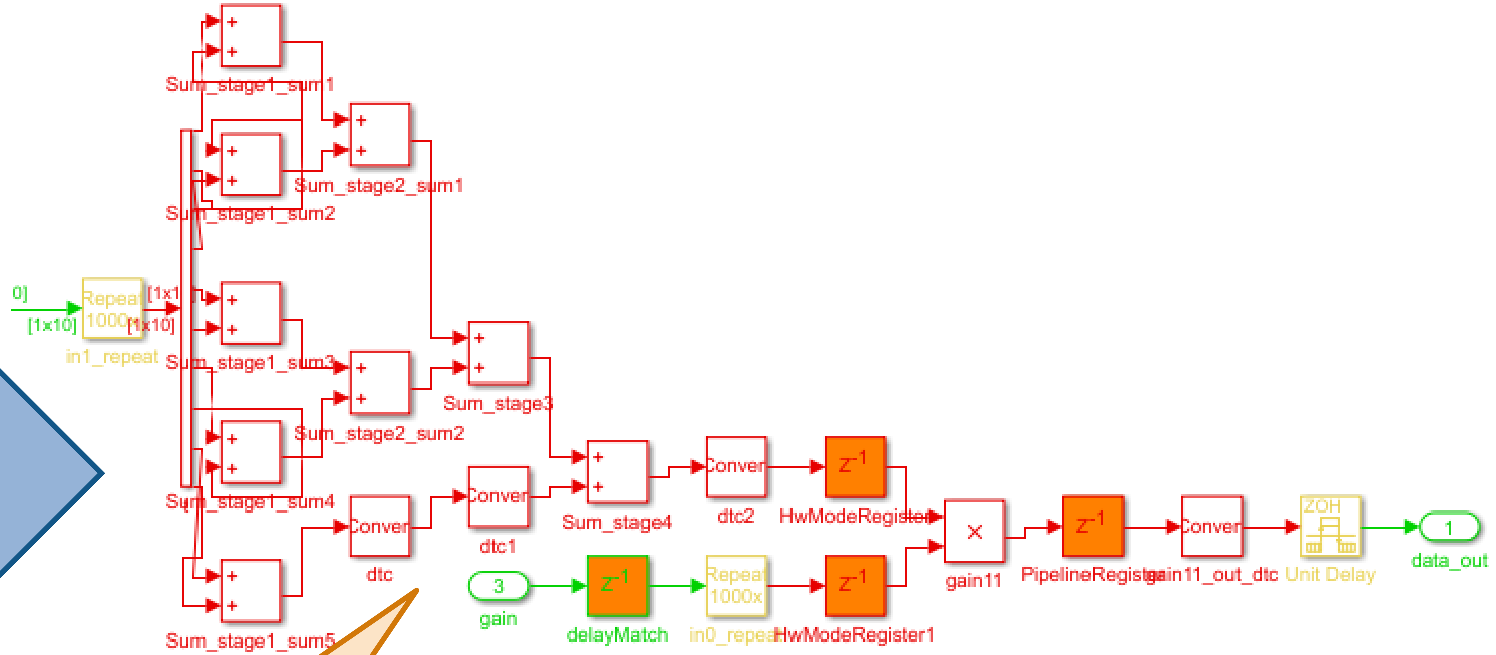
Id	Propagation (ns)	Delay (ns)	Block Path
1	0.2980	0.2980	Unit Delay
2	6.5220	6.2240	a(2)(1)
3	6.5220	0.0000	SumA21
4	6.5220	0.0000	SumA31
5	8.0380	1.5160	SumB31
6	14.2620	6.2240	gain
7	14.2620	0.0000	t
8	15.8840	1.6220	Sum_stage1_sum1
9	17.5060	1.6220	Sum_stage2_sum1
10	19.1280	1.6220	Sum_stage3

Direct traceability between critical path and model



Automatic Pipeline Insertion
Timing Optimization

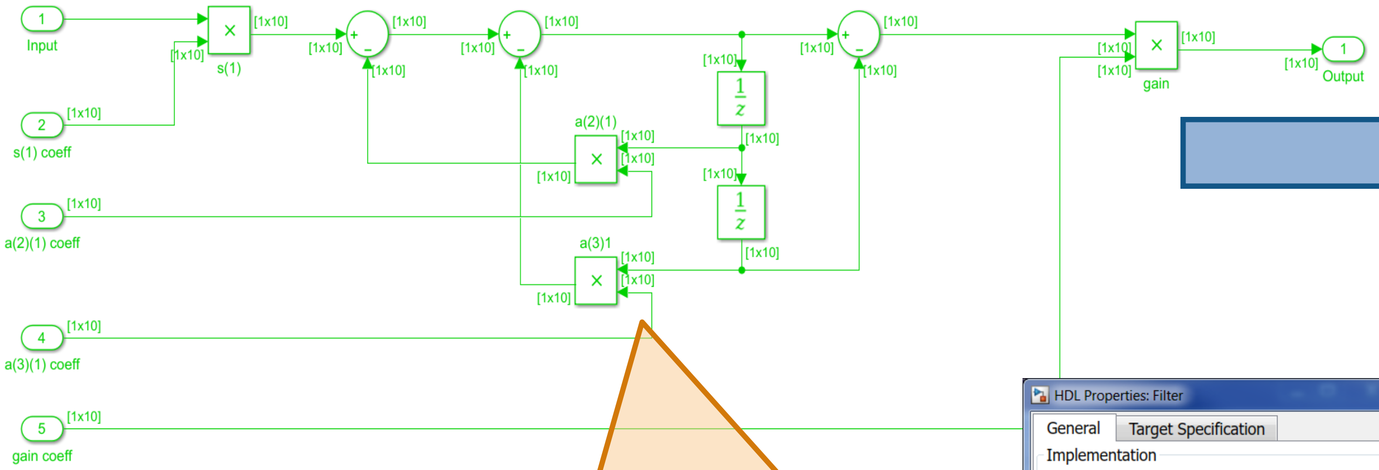


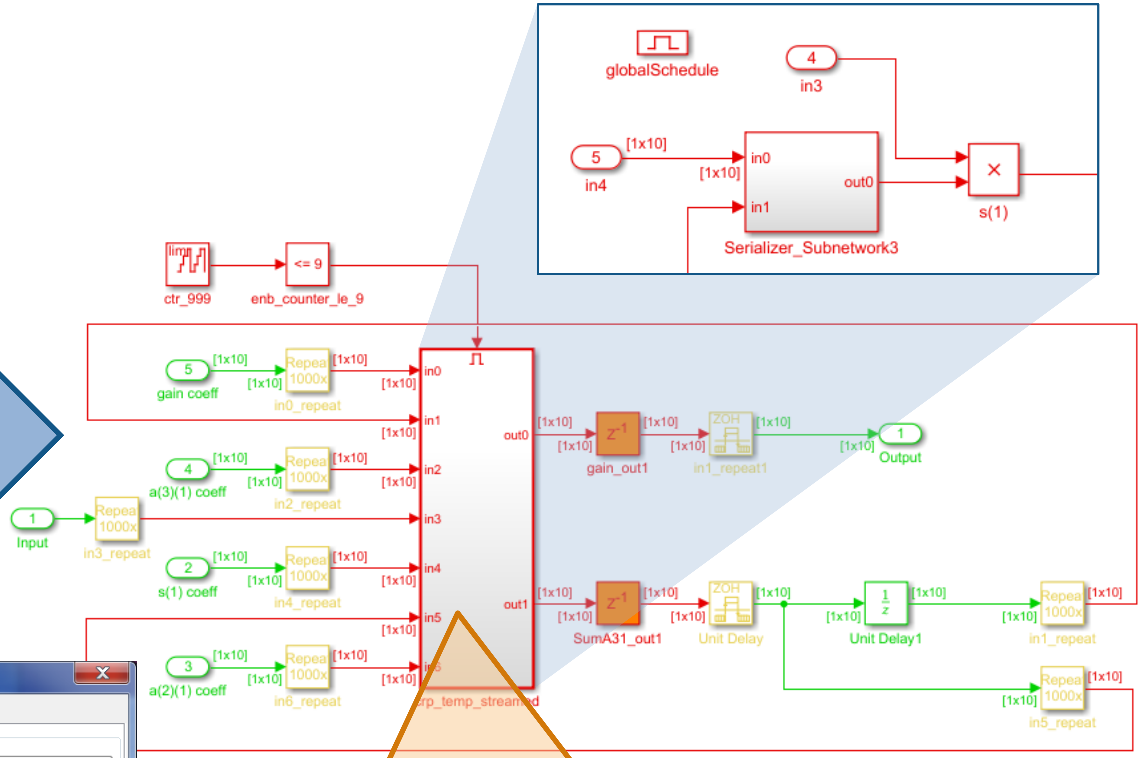


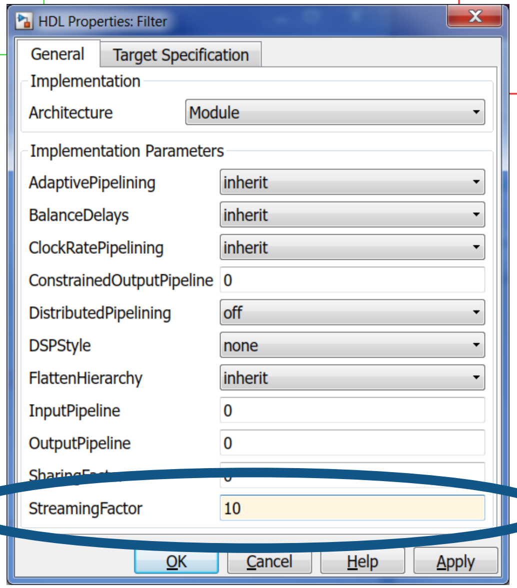
Insert pipelining where it makes sense

Automatic Resource Sharing

Area Optimization







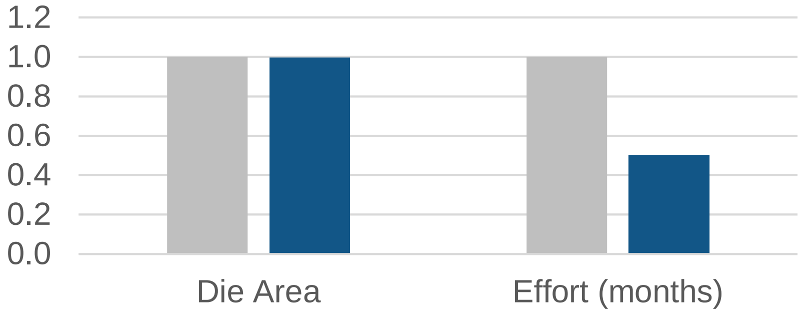
- <u>10 parallel filterbanks</u> implemented
- Data-rate = 100kHz, clk = 100MHz
- Multipliers are utilized at <u>100kHz</u>

- <u>1 streaming filterbank</u> implemented
- Data-rate = 100kHz, clk = 100MHz
- Multipliers are utilized at <u>1MHz</u>

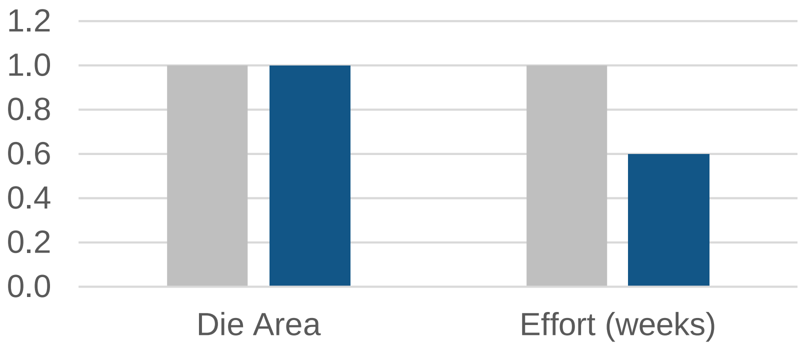
Achieve results with less manual effort

Qualcomm India

Wide-band chain from front-end receiver ASIC



Decimation FIR filter ASIC



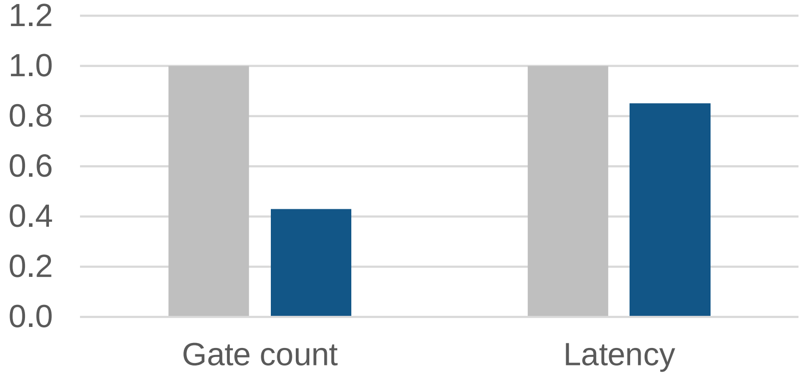
Manual coding

Simulink + HDL Coder

All numbers normalized to 1.0 for "Manual coding"
Smaller numbers are better

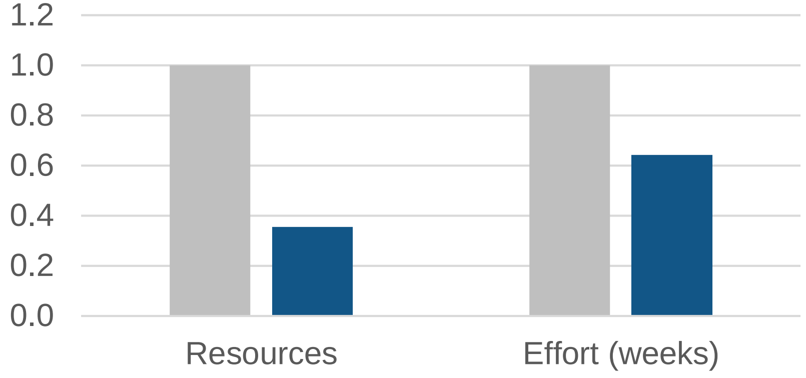
Faraday

Flash NAND Controller ASIC



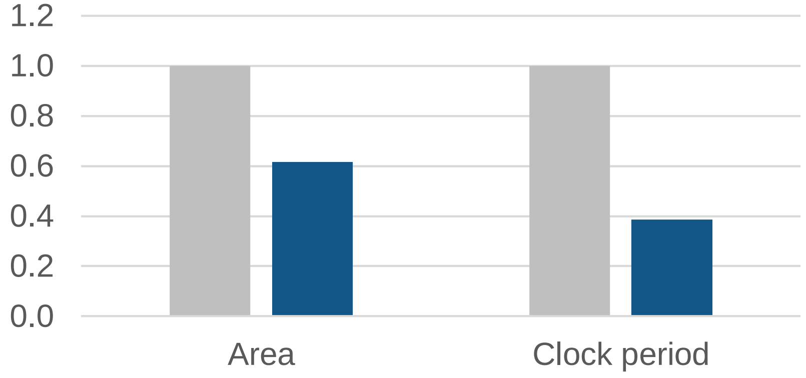
IFM Engineering

3D time-of-flight camera FPGA

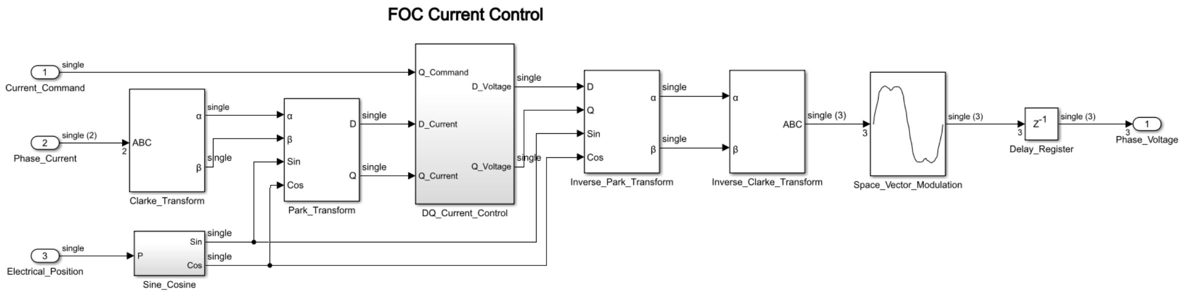


Nokia

FPGA prototype of an ASIC



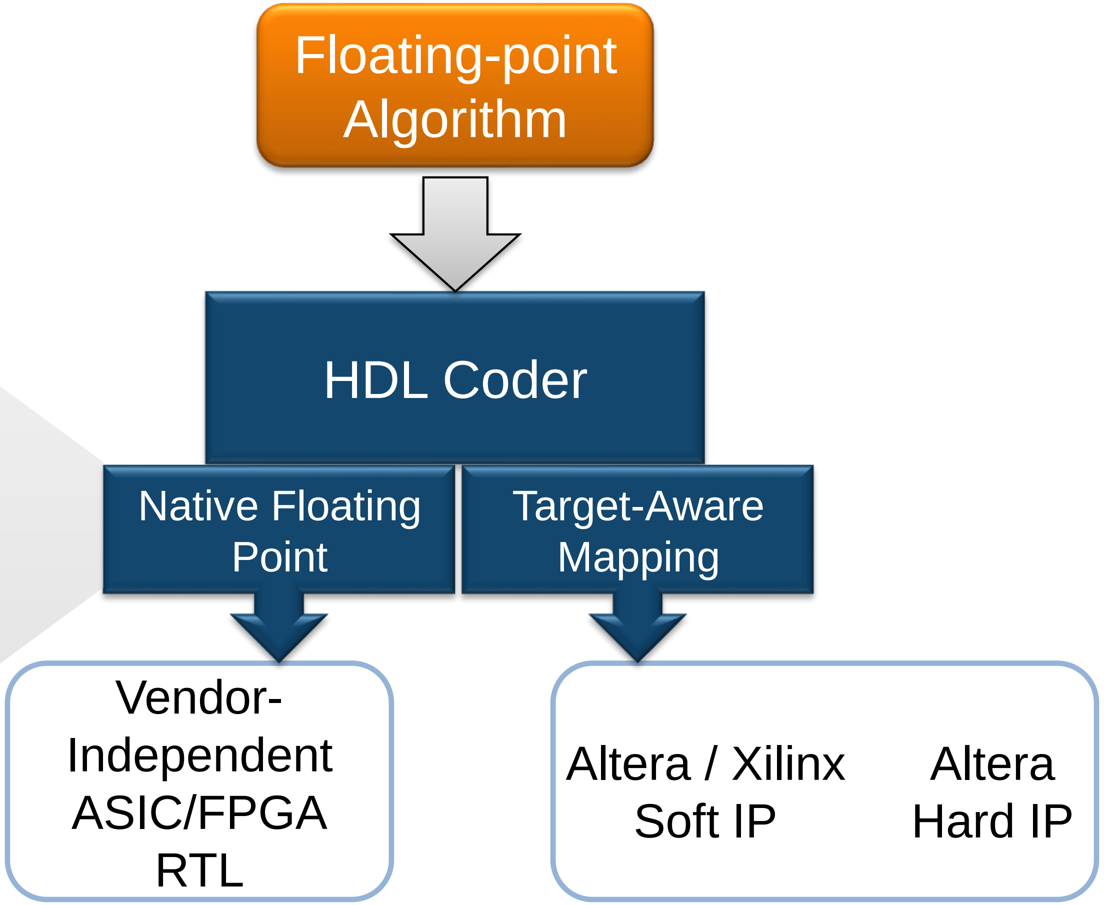
HDL Coder Floating Point Implementation Options



Native Floating Point

- IEEE-754 Single precision support
- Extensive math and trigonometric operator support
- Highly optimal implementations without sacrificing numerical accuracy
- Mix floating and fixed point operations in the same design

embeddedaward2017



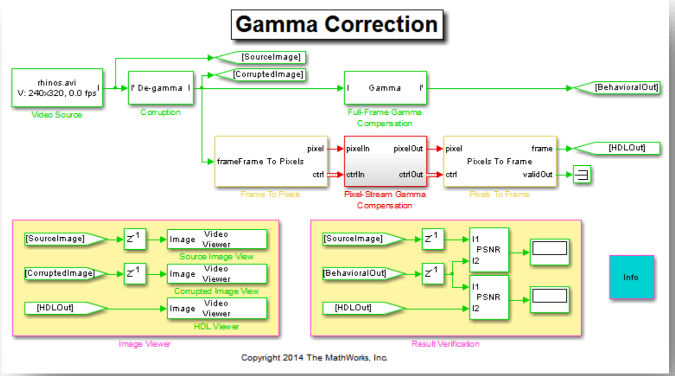
Using HDL Optimized IP Blocks

Image Processing IPs

Design and prototype video image processing systems

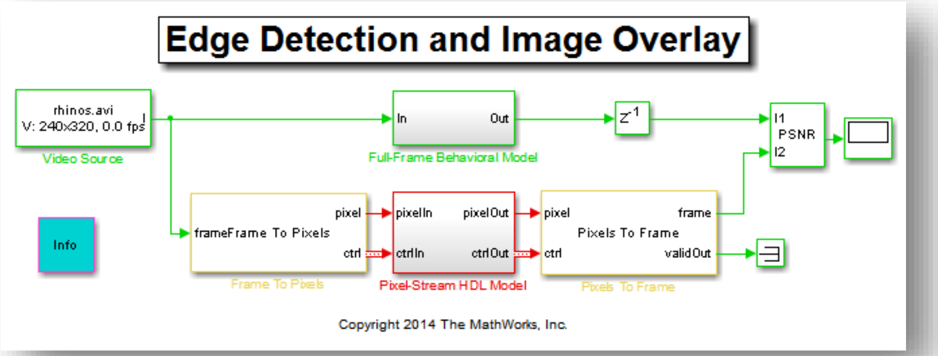
- Modeling hardware behavior of the algorithms
  - Pixel-based functions and blocks
  - Conversion between frames and pixels
  - Standard and custom frame sizes

- Prototyping algorithms on hardware
  - Efficient and readable HDL code
  - FPGA-in-the-loop testing and acceleration

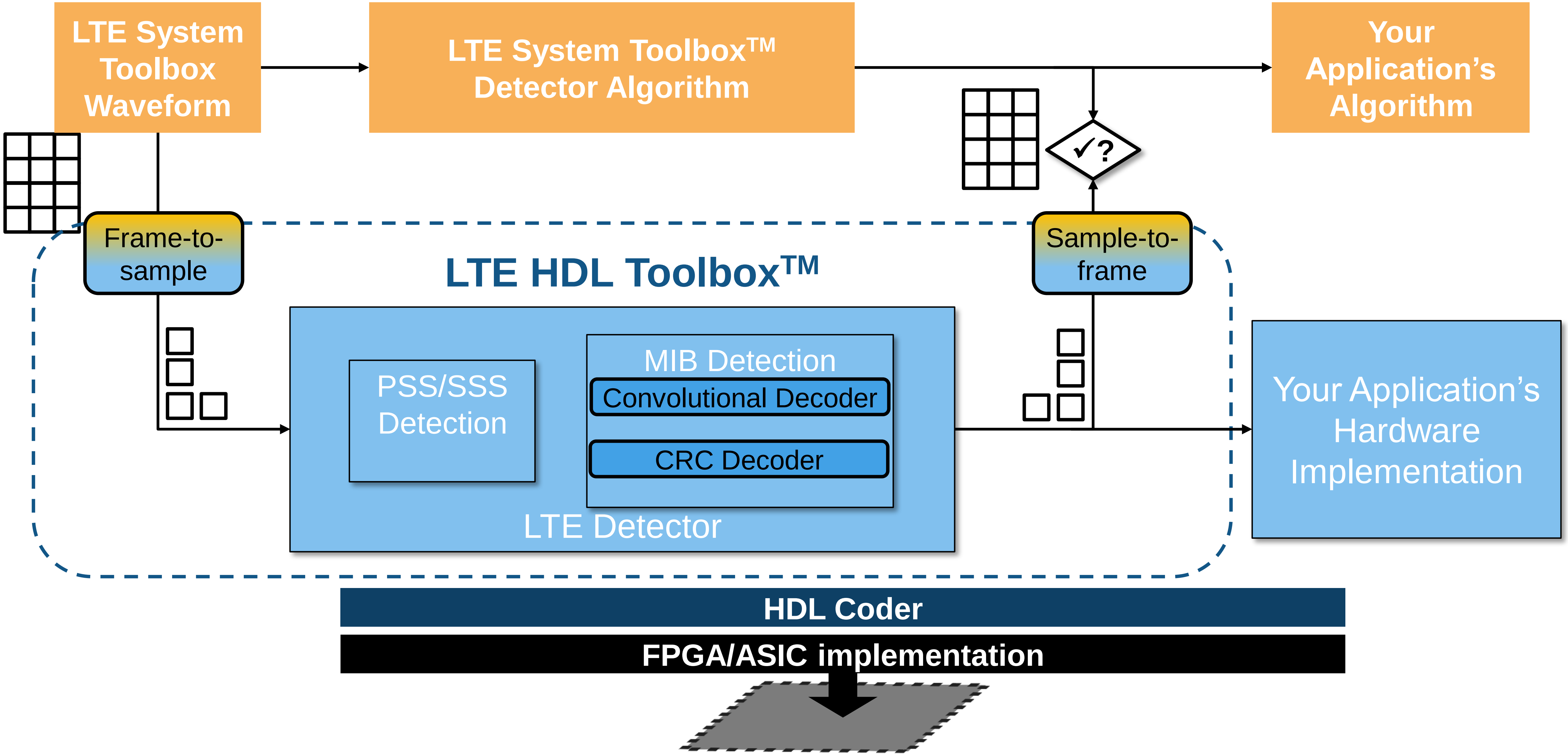


Pixel Based Video Algorithm Library

- **Analysis & Enhancement**
  – Edge Detection, Median Filter
- **Conversions**
  – Chroma Resampling, Color-Space Converter
  – Demosaic Interpolator, Gamma Corrector, Look-up Table
- **Filters**
  – Image Filter, Median Filter, Bilateral Filter
- **Morphological Operations**
  – Dilation, Erosion,
  – Opening, Closing
- **Statistics**
  – Histogram
  – Image Statistics
- **I/O Interfaces**
  – Frame to Pixels, Pixels to Frame, FIL versions
- **Utilities**
  – Pixel Control Bus Creator
  – Pixel Control Bus Selector

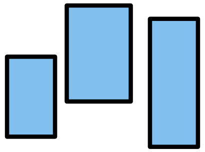


Implementing LTE algorithms

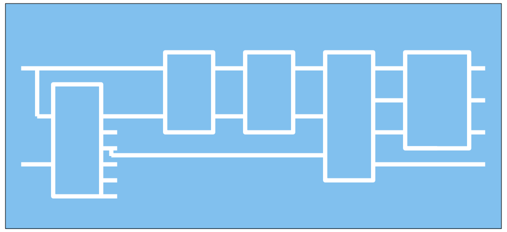


LTE IPs and models

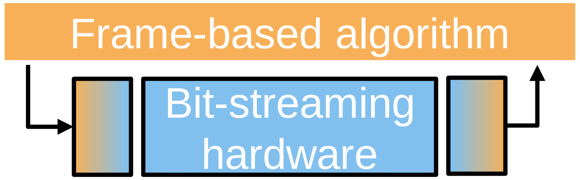
LTE HDL-optimized blocks



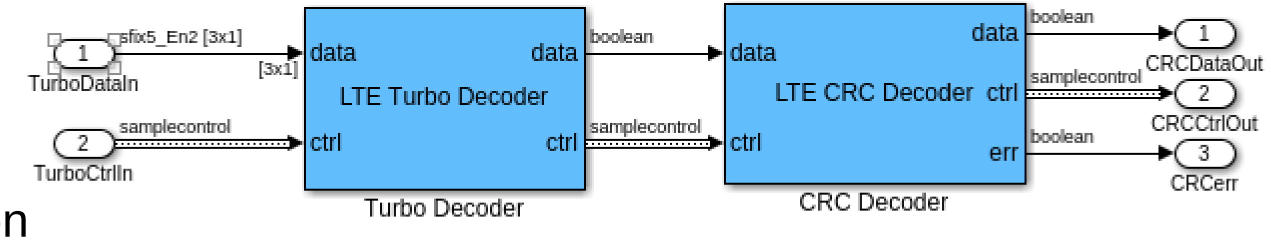
Reference applications



Utilities



- Turbo Encoder
- Turbo Decoder
- Convolutional Encoder
- Convolutional Decoder
- CRC Encoder
- CRC Decoder



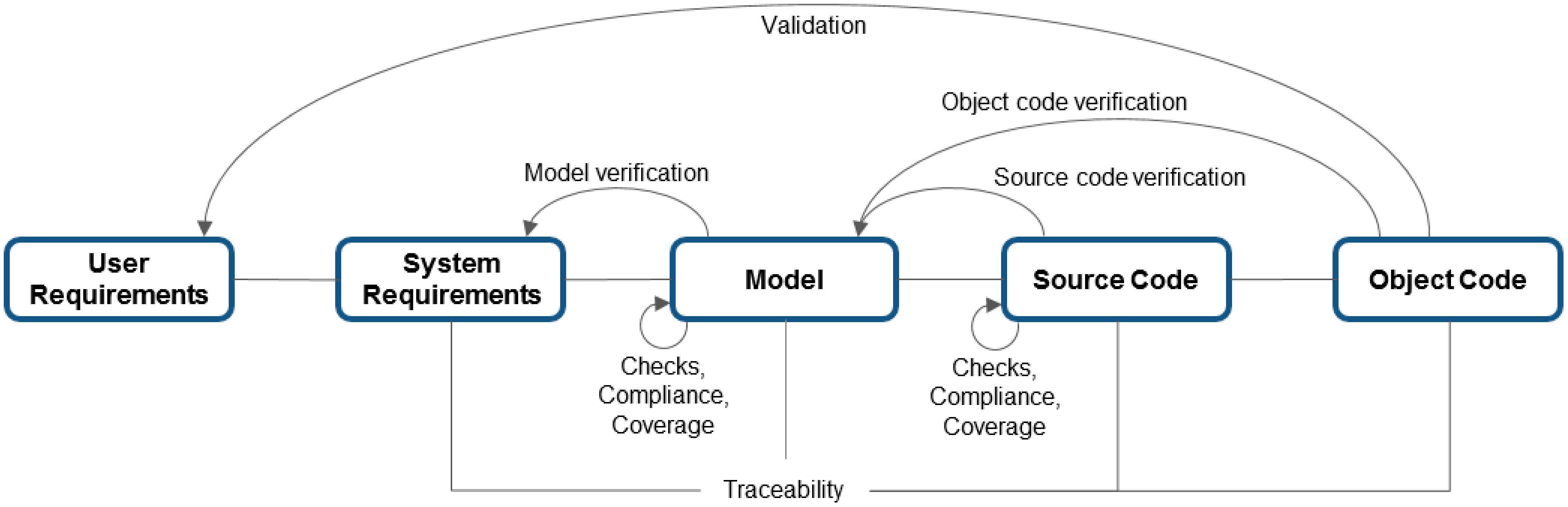
- PSS/SSS Detection
- MIB Recovery
- SIB1 Detection (post-17b)
- LTE Frequency Scanner

- Frame-to-samples / samples-to-frame
- Sample bus creator / selector
- Templates to connect MATLAB tests/golden reference to Simulink HW implementation

Other “HDL Optimized” Algorithms

- Filering
  - Biquad
  - Interpolator/Decimator
  - LMS
- Wireless communications
  - FFT, NCO
  - QAM, BPSK, QPSK
  - Viterbi, Convolutional, RS, Turbo
- Deep learning

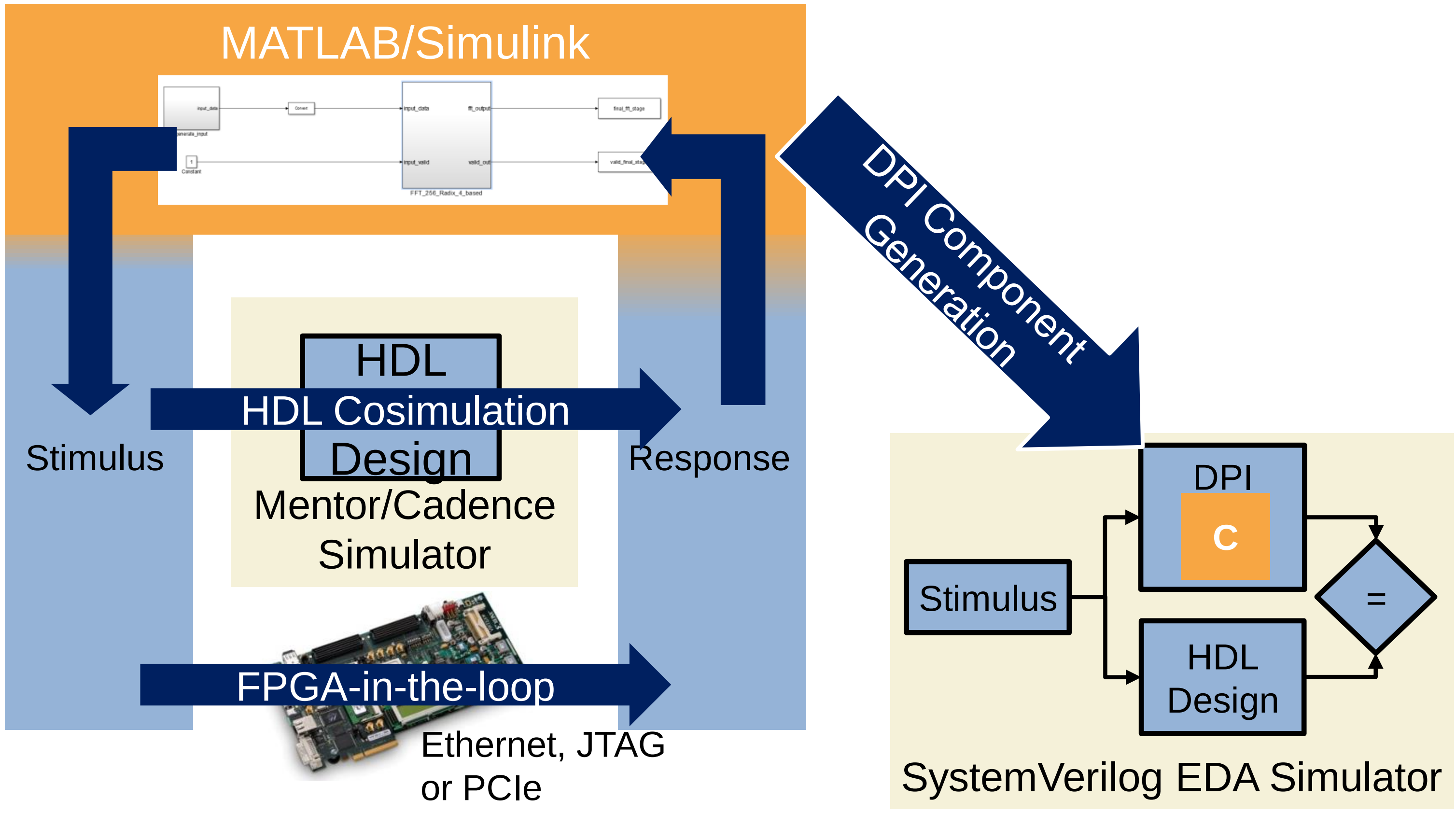
Verification & Validation



- Create a high-level system model and link to requirements documents
- The system model can serve as an executable specification and basis for more detailed models
- Validate requirements by executing your system prototype connected to user interfaces or plant hardware

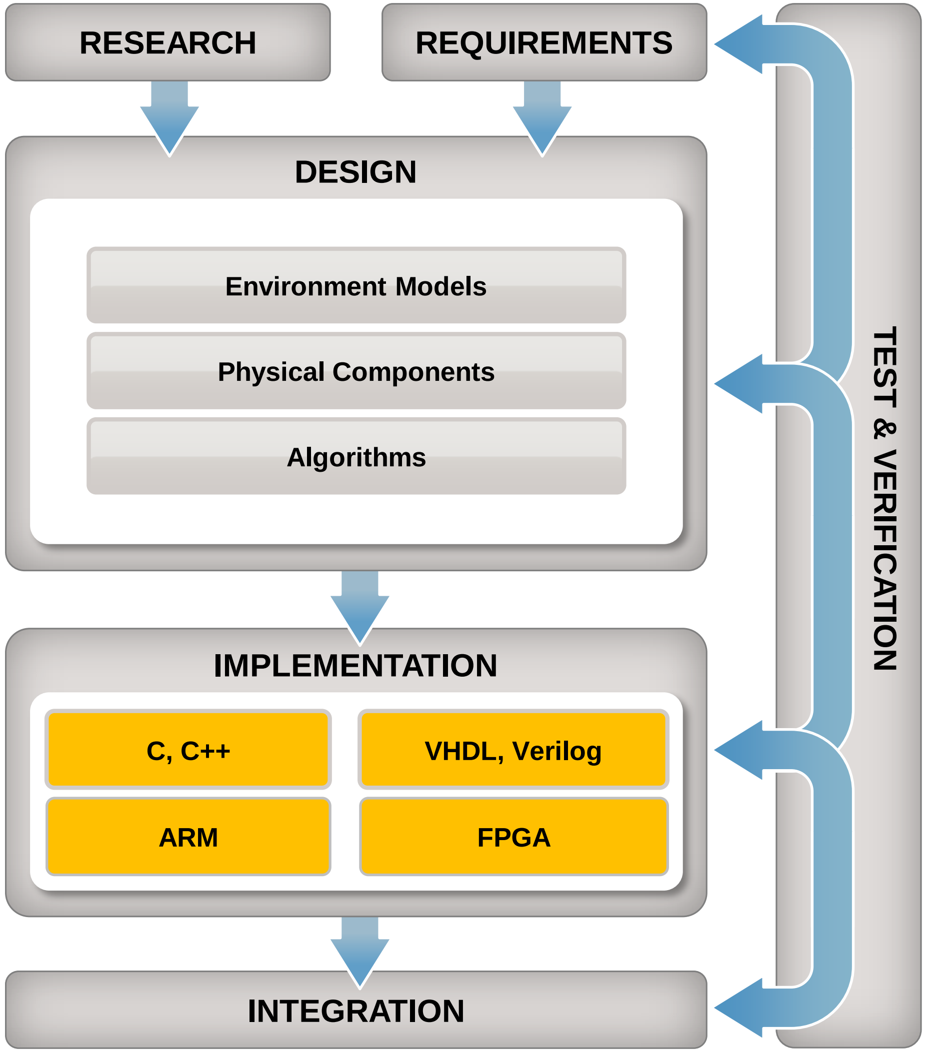
HDL Verifier

Reuse MATLAB/Simulink to automate verification setup



Model-Based Design

From Concept to Production



- Model multi-domain systems
- Explore and optimize system behavior in floating point and fixed point
- Collaborate across teams and continents

- Generate efficient code
- Explore and optimize implementation tradeoffs

- Automate regression testing
- Detect design errors
- Support certification and standards