

# Project-Based Learning for Signal Processing and Communications with MATLAB and Simulink



Houman Zarrinkoub, PhD.

[houmanz@mathworks.com](mailto:houmanz@mathworks.com)



# Agenda

- Project-based learning & signal processing
- Case studies
  - Signal analytics and classification
  - Heart rate monitoring: from algorithms to prototyping on ARM® hardware
  - Over-the-air testing of wireless waveforms with software-defined radios (SDR)
- Summary



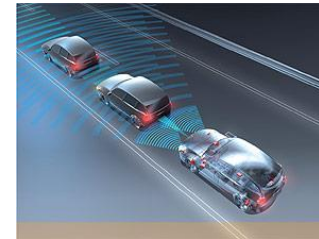
# Project-based learning

- Learning-by-doing
- Deepens understanding
- Excites & motivates
- Fosters collaboration



# Project-based learning & signal processing

- Connected everything
- Live signals available
- Mobile internet everywhere
- Algorithms in everything
  
- Easily design projects that involve signal processing & communications



1

**Signal analytics & classification on data from wearable and mobile devices**

2

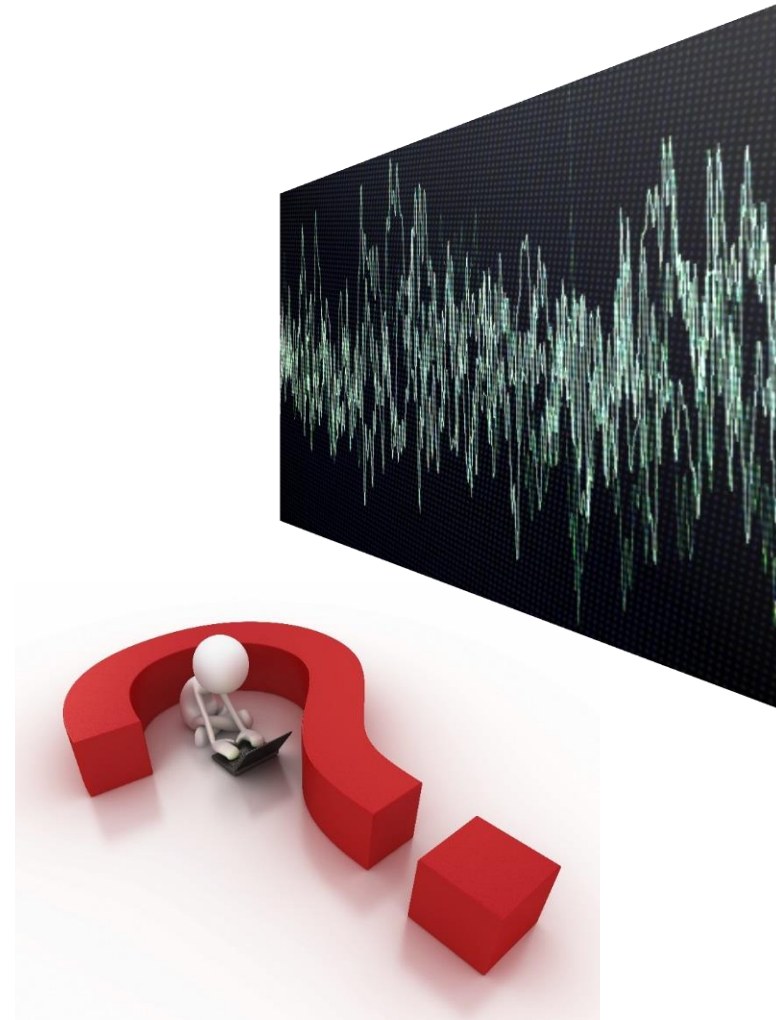
Heart rate monitoring - from desktop processing to prototyping on ARM® Cortex®-M & Cortex®-A

3

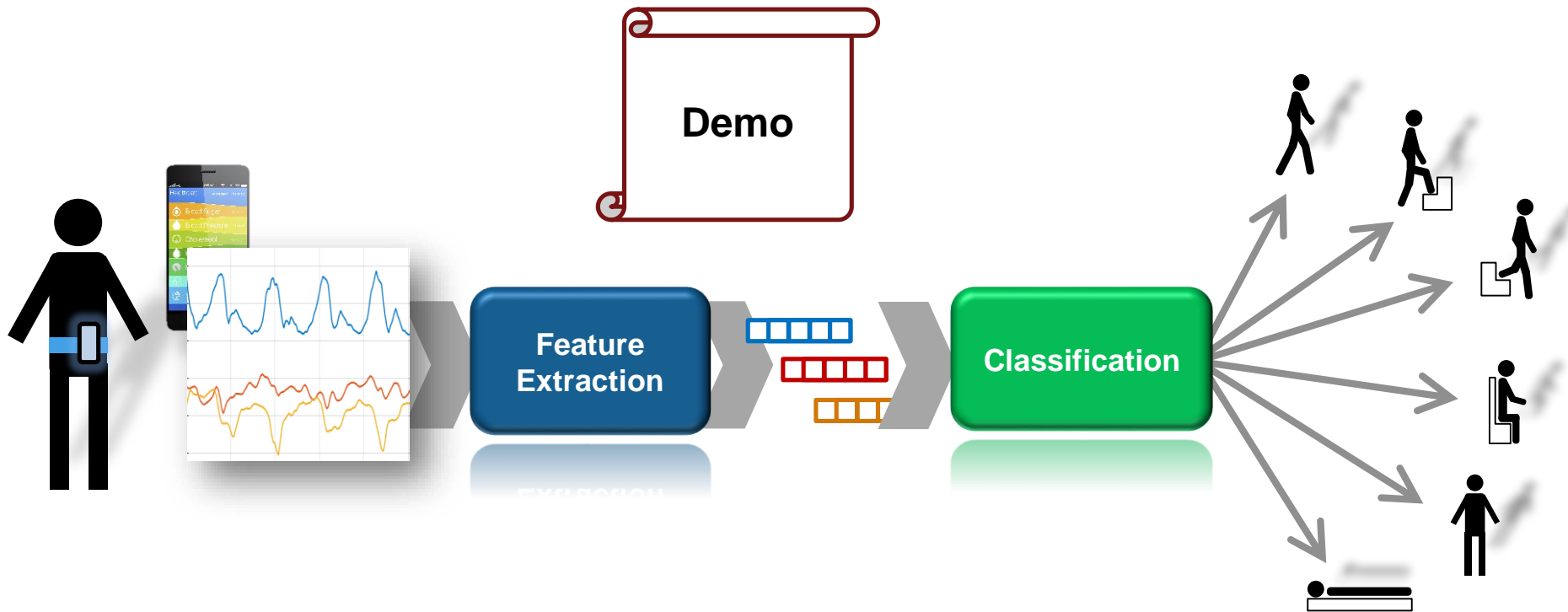
Over-the-air testing of wireless signals – Software-defined radios (SDR) & RF instruments

# Challenges with Signal Analytics

- Data analytics on sensor signals
- Need domain-specific knowledge
- Open-ended problem and long discovery cycles
- You can use offline pre-recorded training and validation data sets



# Example: Human Kinetic Activity Analysis/Classification



## Dataset courtesy of:

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra and Jorge L. Reyes-Ortiz.

*Human Activity Recognition on Smartphones using a Multiclass Hardware-Friendly Support Vector Machine.*

International Workshop of Ambient Assisted Living (IWAAL 2012). Vitoria-Gasteiz, Spain. Dec 2012

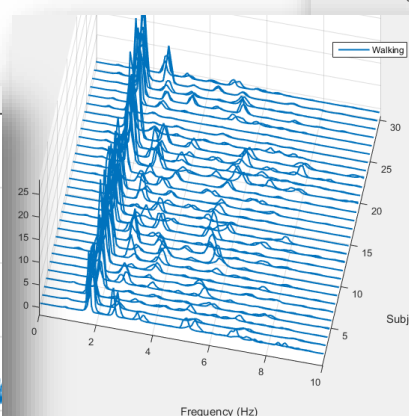
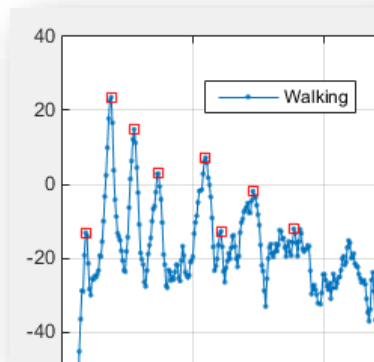
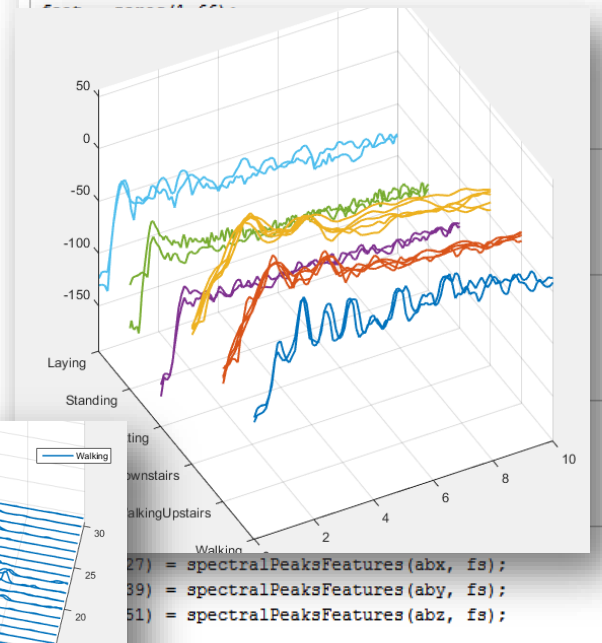
<http://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones>

# Human Kinetic Activity Analysis/Classification Solution

- Only core built-in Signal Processing algorithms
- 66 high-quality features extracted with only 65 lines of code!
- Visualization and automation accelerate insight iterations

```
function feat = featuresFromBuffer(atx, aty, atz, fs)

persistent flp fhp
if isempty(flp)
    f1 = 0.4;
    f2 = 0.8;
    flp = lpfilter(f1,f2);
    flp.PersistentMemory = false;
    fhp = hpfilter(f1,f2);
    fhp.PersistentMemory = false;
end
```





# Leverage built-in algorithms

## Do you want to re-invent the wheel?

- **Signal Processing Toolbox 7.0**

- Simplified interface and examples for casual users
- Analysis of non-evenly sampled data (IoT)
- New functions for signal measurements, filter design

- **Neural Network Toolbox**

- **Statistics & Machine Learning Toolbox**

- cheby2
- filter
- rms
- pwelch
- periodogram
- xcov
- findpeaks
- ...

$$|H(j\omega)|^2 = \frac{\epsilon^2 C_N^2(\omega_s/\omega)}{1 + \epsilon^2 C_N^2(\omega_s/\omega)}$$

$$C_N(\omega_s/\omega) = \begin{cases} \cos[N \cos^{-1}(\omega_s/\omega)] , & |\omega| \geq \omega_s \\ \cosh[N \cosh^{-1}(\omega_s/\omega)] , & |\omega| \leq \omega_s \end{cases}$$

$$P_{xx}(f) = \frac{1}{LF_s} \left| \sum_{n=0}^{L-1} x_L(n) e^{-j2\pi f n / F_s} \right|^2$$

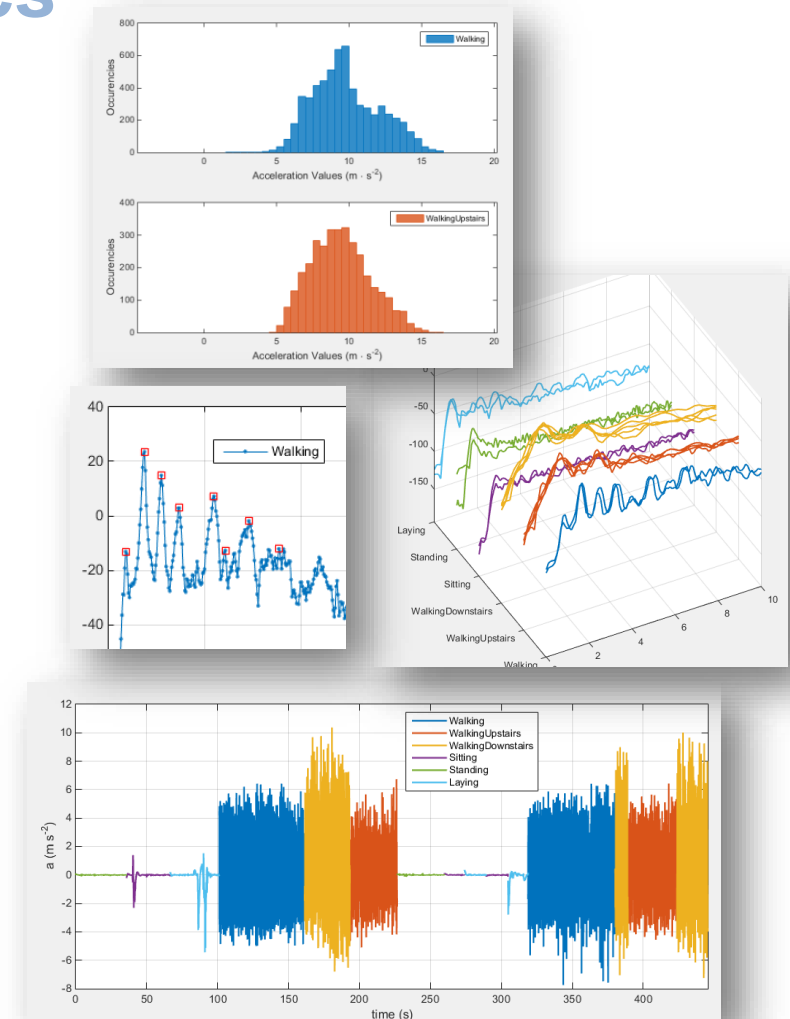
$$f_k = \frac{kF_s}{N} \quad k=0,1,\dots,N-1$$

$$c_{xy}(m) = \begin{cases} \sum_{n=0}^{N-|m|-1} \left( x(n+m) - \frac{1}{N} \sum_{i=0}^{N-1} x_i \right) \left( y_n^* - \frac{1}{N} \sum_{i=0}^{N-1} y_i^* \right) & m \geq 0 \\ c_{yx}^*(-m) & m < 0 \end{cases}$$

# Summary

## MATLAB for Signal Analytics

- Signal Processing Toolbox has an extensive set of functions for signal processing and analysis
- Visualisation and App-driven automation accelerate insight iterations
- All this comes with compact & concise MATLAB language and extensive documentation
- Applicable to Internet-of-Things (IoT) projects



1

Signal analytics & classification on data from wearable and mobile devices

2

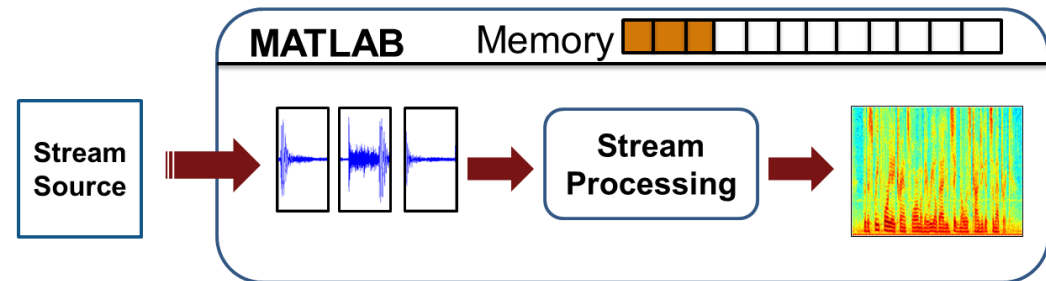
Heart rate monitoring - from desktop processing to prototyping on ARM® Cortex®-A

3

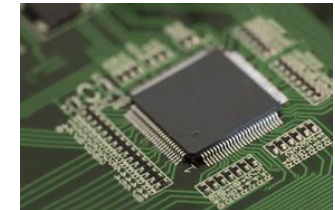
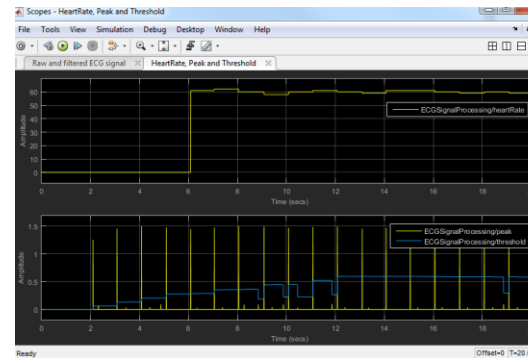
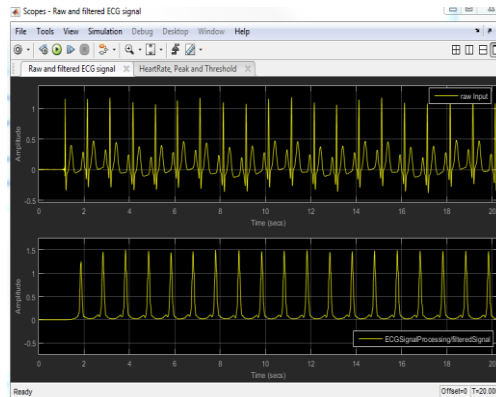
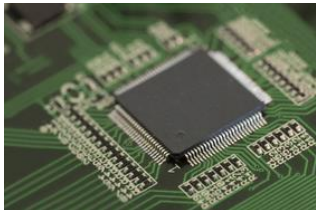
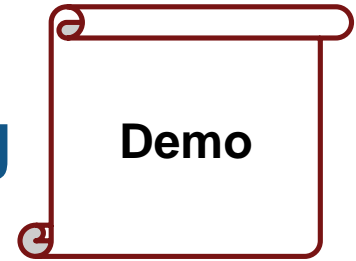
Over-the-air testing of wireless signals – Software-defined radios (SDR) & RF instruments

# Challenges with streaming DSP

- Stream real-world data into and out of MATLAB
- Algorithms must keep up with rate of incoming frames of data
- Need domain-specific knowledge for both algorithms and data I/O



# Example: Smart Sensor for Heart Rate Monitoring



Acquire streaming sensor data (ECG) in real-time with BeagleBone Black (BBB) using ARM Cortex-A

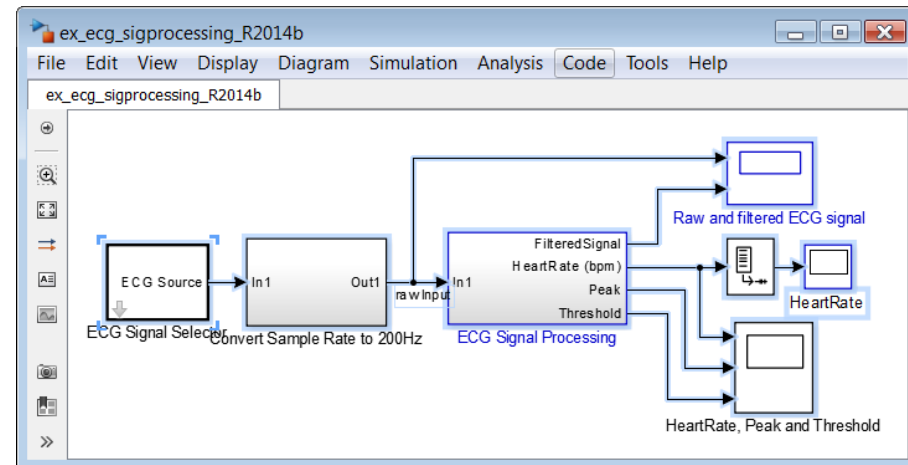
Streaming algorithms: filtering, sample rate conversion, detection and classification in real-time

Prototype: Test algorithms using External-mode

Implement and deploy on ARM Cortex-A BBB board

# Smart Sensor for Heart Rate Monitoring Solution

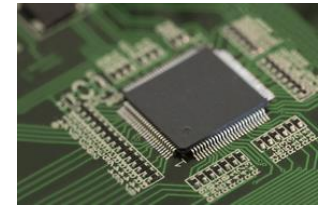
- A Model-Based Design Workflow
- Executable specification & simulation: pre process, filter & detect ECG Signal
- Code Generation: Create a PIL block with optimized code on ARM Cortex-A processor
- Verification: Perform real-time processing with BeagleBone Black



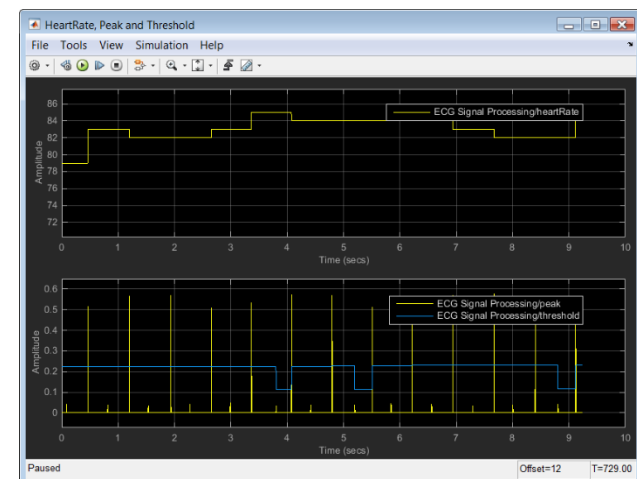
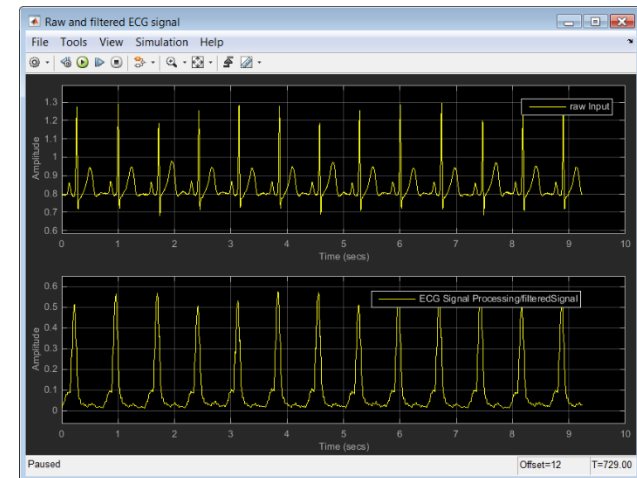
<p><b>Contents</b></p> <ul style="list-style-type: none"> <li><a href="#">Summary</a></li> <li><a href="#">Subsystem Report</a></li> <li><a href="#">Code Interface Report</a></li> <li><a href="#">Traceability Report</a></li> <li><a href="#">Static Code Metrics Report</a></li> <li><a href="#">Code Replacements Report</a></li> </ul> <p><b>Generated Code</b></p> <ul style="list-style-type: none"> <li>[ - ] Main file <ul style="list-style-type: none"> <li><a href="#">ert_main.c</a></li> </ul> </li> <li>[ - ] Model files <ul style="list-style-type: none"> <li><a href="#">ECGSignalProcessingSubsystem.c</a></li> <li><a href="#">ECGSignalProcessingSubsystem.h</a></li> <li><a href="#">ECGSignalProcessingSubsystem_U.c</a></li> <li><a href="#">ECGSignalProcessingSubsystem_U.h</a></li> </ul> </li> </ul>	<pre> 157 real_T y; 158 int32_T i; 159 int16_T RRSum; 160 uint32_T tmp; 161 162 /* DiscreteFir: '&lt;Si&gt;/Bandpass_Filter' incorporates: 163  * Inport: '&lt;Root&gt;/In1' 164  */ 165 ne10_fir_float_neon(&amp;ECGSignalProcessingSubsystem_DW.S, 166                    &amp;ECGSignalProcessingSubsystem_U.rawInput[0], 167                    &amp;ECGSignalProcessingSubsystem_B.BandpassFilter[0], 50U); 168 169 /* DiscreteFir: '&lt;Si&gt;/derivative' */ 170 ne10_fir_float_neon(&amp;ECGSignalProcessingSubsystem_DW.S_c, 171                    &amp;ECGSignalProcessingSubsystem_B.BandpassFilter[0], 172                    &amp;ECGSignalProcessingSubsystem_B.Abs[0], 50U); 173 174 /* Abs: '&lt;Si&gt;/Abs' */ 175 for (i = 0; i &lt; 50; i++) { 176     ECGSignalProcessingSubsystem_B.Abs[i] = (real32_T) fabs 177     (ECGSignalProcessingSubsystem_B.Abs[i]); 178 } </pre>
--	--

# Summary

## MATLAB/Simulink for Streaming DSP



- DSP System Toolbox provides streaming capabilities for design and simulation of real-time sensor processing in MATLAB and Simulink
- Once your design works in your desktop, you can deploy it to Low-cost Hardware such as BeagleBone Black, Raspberry Pi, etc.
- With Embedded Coder you can configure your embedded application and implement, test and validate on hardware



1

Signal analytics & classification on data from wearable and mobile devices

2

Heart rate monitoring - from desktop processing to prototyping on ARM® Cortex®-M & Cortex®-A

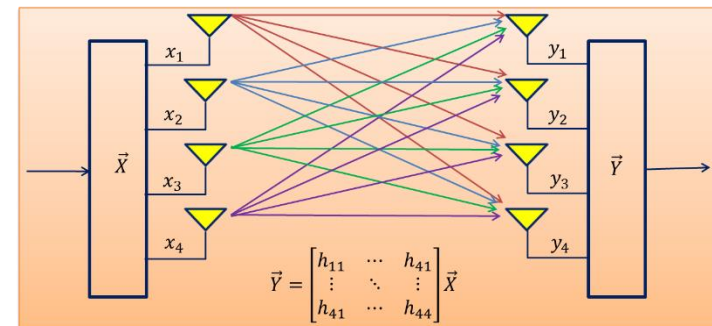
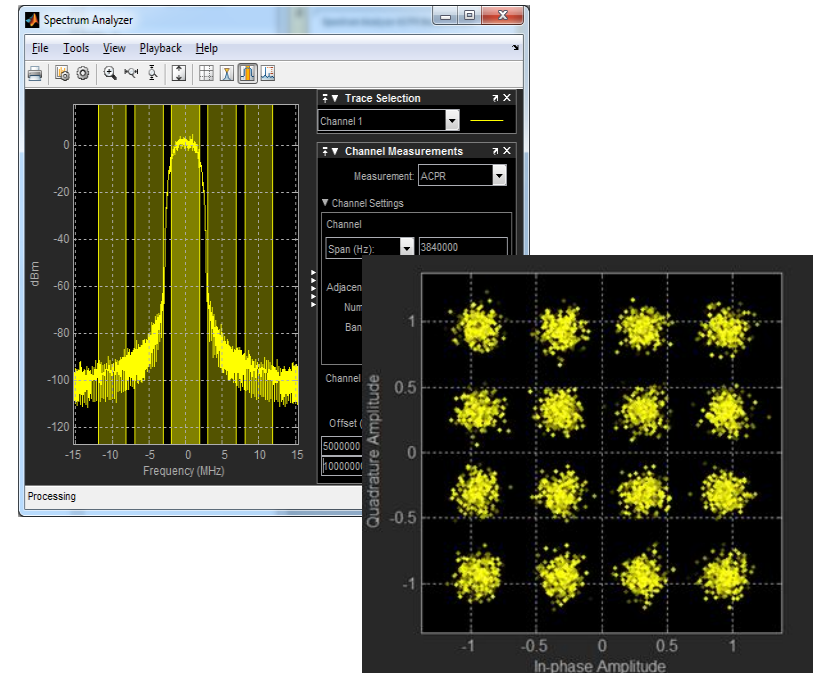
3

**Wireless Over-the-air testing –  
Software-defined radios (SDR) & RF instruments**

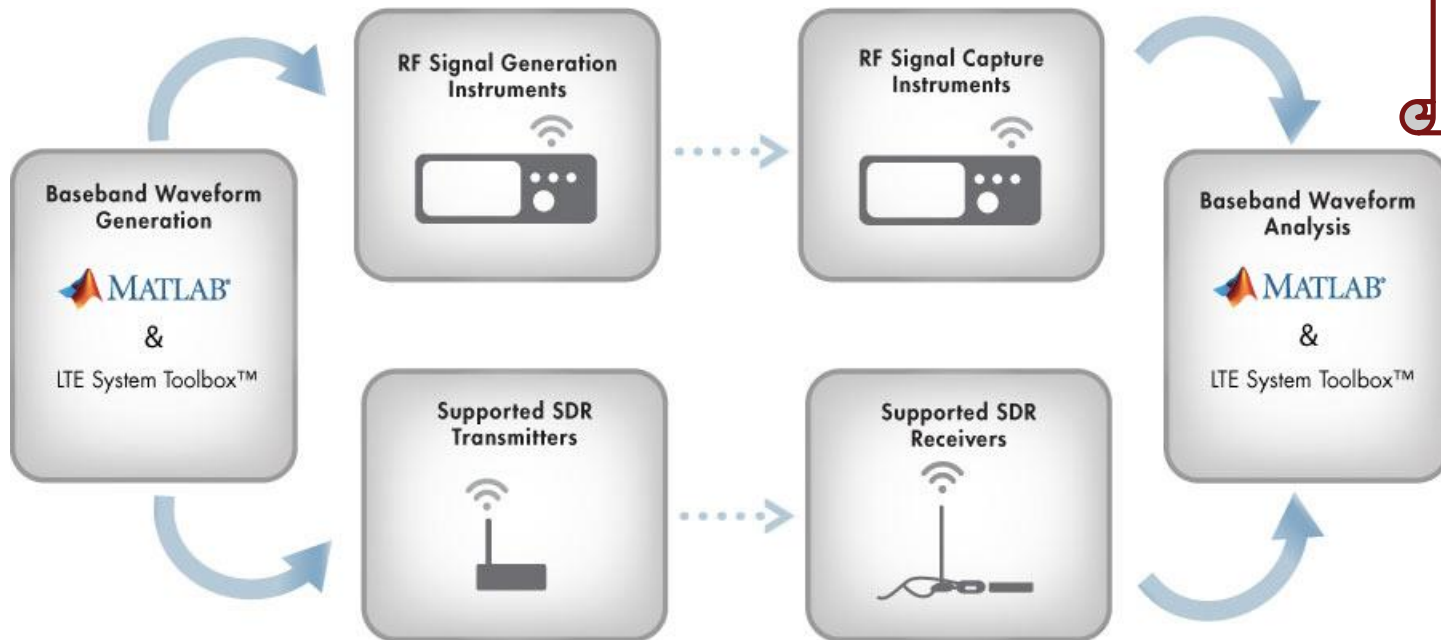


# Challenges of Wireless Communications

- Need multi-domain (Digital, RF, Antenna) knowledge
- Jointly optimize Digital Baseband, RF circuits and antenna patterns
- Simulate first with channel modelling. Is your channel mode accurate?
- Test it with over-the-air transmission and reception of RF signals



# Example: Over-the-air testing with SDRs & RF instruments



Process original data bits and generate custom digital baseband waveforms in transmitter

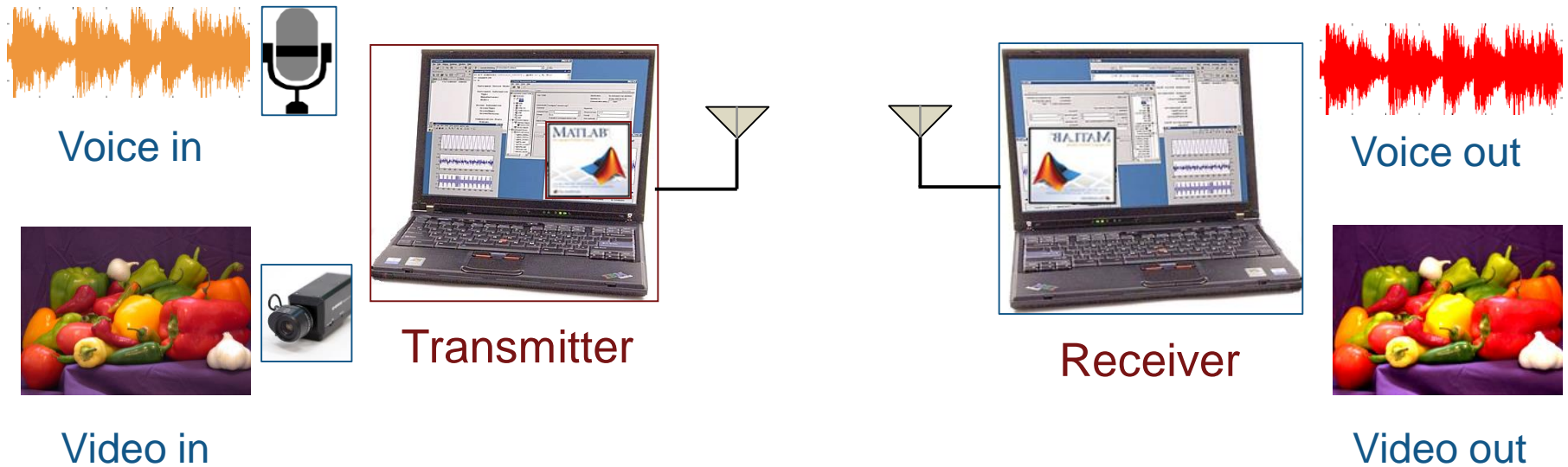
Transmit waveform using SDR devices or RF instruments

Capture received samples with SDR devices or RF instruments

Process received samples in receiver. Decode/recover original data

# Over-the-air testing with SDRs & RF instruments

## Solution



# Supported SDRs & RF instruments

RF Signal Generator



Zynq Radio SDR



USRP SDR



RF Spectrum Analyzer



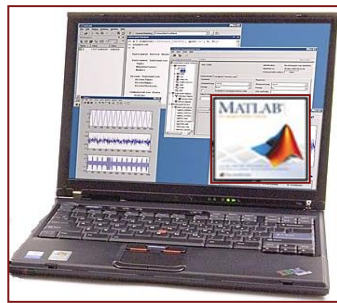
Zynq Radio SDR



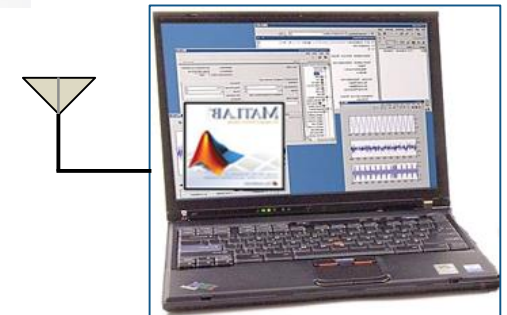
USRP SDR



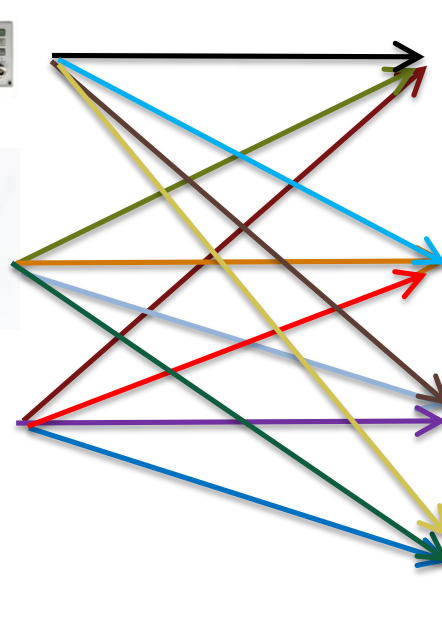
RTL SDR



Transmitter



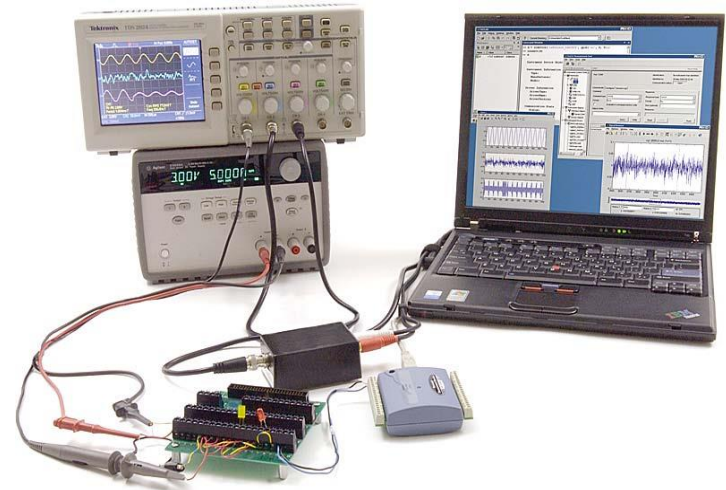
Receiver



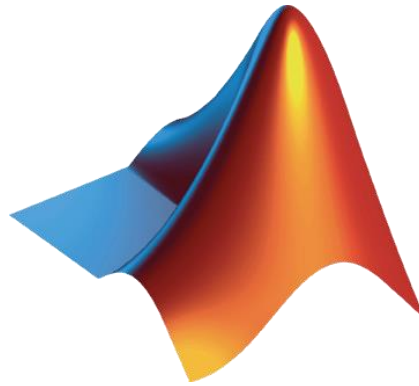
# Summary

## MATLAB/Simulink for Radio Connectivity

- Go beyond simulation
- Actually, transmit & receive wireless signals with MATLAB and Simulink
- Connect to SDRs
  - Communications System Toolbox hardware support packages
- Connect to RF instruments
  - Instrument Control Toolbox
- Perform over-the-air testing
- Verify your wireless designs



Thank You



Q & A