Modelling and Prototyping Signal Processing and Communications Systems

Graham Reith
Industry Manager: Communications, Electronics & Semiconductors
Session Outline

- **Part 1 – Algorithm Development**
  - Developing signal processing algorithms with relevant toolboxes

- **Part 2 – System Modelling**
  - Simulating algorithms with real-world implementation effects

- **Part 3 – Prototyping Systems and SDR Platforms**
  - Developing prototypes to test with real-world signals
Workflow to go from an idea to a product can be broken into 5 activity types:

- Aim to establish a natural connection between all these different stages in the product development workflow
  - Enables new ideas to flow from concept into the product, without undue hindrance
  - Promotes innovation and enables agility
MATLAB Example: Parametric Audio Equalizer

Guitar10min.ogg a 44.1Khz stereo audio

Custom Audio Algorithm

Parameter Equalizer Filters

Tune parameters in real-time

Tune it

Create it

Play it

See it

Array Plot

Hear it

Speaker

Visualize audio waveforms in real-time

MIDI Control
Introducing System Objects

- 2 Entry Points for describing Signal Processing operations in MATLAB:
  - Data analysis
  - Algorithm exploration
  - System simulation
  - Efficient implementation

```matlab
% Apply filter to data all at once
output = filter(b,a,data);

% Or, save and restore state each call
[output,state] = filter(b,a,data,state);

% Instantiate filter with coefficients
Hf = dsp.FIRFilter('Numerator',b);

% Process data through filter
output = step(Hf,data);
```
Challenges in Signal Processing System Design

Framework for Real-Time Simulations

“I have to process large data and test my simulations with streaming signals. I need a simulation testbench that can keep up with real-time data.”

Quick Innovation & Modelling

“I need to find innovative algorithms and create and model a working system very quickly.”

Rapid Prototyping & Simulation Acceleration

“I need to optimize my high-level MATLAB algorithm for speed. I then need to verify that the optimized code works the same way as the original MATLAB code.”
I have to process large data and test my simulations with streaming signals. I need a simulation testbench that can keep up with real-time data.

I need to find innovative algorithms and create and model a working system very quickly.

I need to optimize my high-level MATLAB algorithm for speed. I then need to verify that the optimized code works the same way as the original MATLAB code.

What DSP System Toolbox Provides in MATLAB

**Framework for Real-Time Simulations**
Stream processing technique and hardware peripheral access that speeds up simulation and reduces memory footprint

**Quick Innovation & Modelling**
Pre-defined algorithms as functions and System Objects for quick prototyping

**Rapid Prototyping & Simulation Acceleration**
Support for C/C++ code generation that enables design continuity and faster simulation
Phased Array System Toolbox

Design and simulate phased array systems
Applications in RADAR, Communications, and Audio

- Arrays, waveforms, targets, clutter, etc...
- Signal processing algorithms

MATLAB EXPO 2015
UNITED KINGDOM
Example: FMCW RADAR

- Antenna Array Analysis
- Simulation of Signal Processing
Progression from Algorithms to Systems

- **WHAT**
  - Initial focus on ‘what’ the algorithm should be

- **HOW**
  - Then need to consider ‘how’ it should be implemented
    - Fixed-Point, analog vs digital methods
    - Architecture trade-offs, implementation cost
  - Requires collaboration between different teams to gain insight, share experience
  - System Modelling provides mechanism for collaboration across different teams
Modelling from Antenna to Bits

Antenna, Antenna arrays
- type of element, # elements, configuration
- Antenna Toolbox
- Phased Array System Toolbox

Channel
- interference, clutter, noise
- Communications System Toolbox
- LTE System Toolbox
- Phased Array System Toolbox

RX
- LNA
- ADC
- SimRF
- RF Toolbox

Mixed-Signal
- Continuous & discrete time
- Simulink (Simscape)
- DSP System Toolbox
- Control System Toolbox

TX
- PA
- DAC
- Phased Array System Toolbox
- Signal Processing Toolbox
- Communications System Tool box
- LTE System Toolbox

Algorithms
- beamforming, beamsteering, MIMO
- Phased Array System Toolbox
- DSP System Toolbox
- Communications System Toolbox
- LTE System Toolbox

RF Impairments
- frequency dependency, non-linearity, noise, mismatches
- DAC
- TX
- ADC
- RX

Waveforms
- Phased Array System Toolbox
- Signal Processing Toolbox
- Communications System Toolbox
- LTE System Toolbox

MATLAB EXPO 2015
UNITED KINGDOM
Example: AD9361 Simulation Model

- Predict the impact of RF Imperfections in test signals
- Use reference tones and LTE Signals
- Evaluate the effects of nonlinearity, noise, gain and phase imbalance
- Generate AD9361 configuration file
LTE System Toolbox

- Standard-compliant models for LTE and LTE-Advanced
- End-to-end physical layer transmit and receive processing functions
- Test models and reference measurement channel waveform generators
Model-Based Design
A single shared development environment
Model-Based Design
A single shared development environment

- Verify operation in simulation
- Validate performance with live signals
- Deploy design on production system
Radio Prototyping Hardware Support

- **RTL-SDR**
  - Ultra low-cost, low-bandwidth
  - Rx Only

- **USRP**
  - Customizable RF front-end

- **Xilinx FPGA-based radio**
  - High bandwidth (~25MHz)
  - Possible FPGA target

- **Spectrum Analyzers**
  - High-quality RF front-end
  - Wide frequency range, high bandwidth
  - Higher cost
Example: RTL-SDR USB RF Receiver

```matlab
% Other parameters. Change this center frequency to survey a different
% band.
% Set initial parameters
fc = 102.5e6; % Center frequency (Hz)
FrontEndSampleRate = 1e6; % Samples per second
FrameLength = 256*20;

% Create receiver and spectrum analyzer System objects
hSSDRx = comm.SDRRTLReceiver(...
    'CenterFrequency', fc, ... 
    'EnableTunerAGC', true, ... 
    'SampleRate', FrontEndSampleRate, ... 
    'SamplesPerFrame', FrameLength, ... 
    'OutputDataType', 'double');

hSpectrum = dsp.SpectrumAnalyzer(...
    'Name', 'Actual Frequency Offset');
```

Sampling frequency (up to ~2.8MHz)
RF Centre Frequency (20MHz to ~1.8GHz)
+ Tuner gain parameters
Frequency correction parameters
Raspberry Pi with RTL-SDR

- New in R2015b: Using the Raspberry Pi with the RTL-SDR receiver
- Deploy signal processing algorithms directly to the ARM processor on the Raspberry Pi, and stream in data from the RTL-SDR receiver
- Output audio, or send data to ThingSpeak, etc
Exhibition: Using SDR Platforms for Communications Prototyping

- Transmit and receive live image / audio on 4G LTE signals using Zynq based Software Defined Radio (SDR) and LTE System Toolbox
- Model the AD9361 RF chip used on the SDR platform
Xilinx FPGA-Based Radio: SDR Peripheral

- Execute fixed radio functions on FPGA
- Tunable pre-defined radio parameters
Xilinx FPGA-Based Radio: SDR Target

- Generate HDL code to implement custom radio functionality on FPGA
Production Integration

Simulation

- Simulink
- Algorithm Model
  - Algorithm Model

Prototype

- ARM
  - Algorithm C
  - Linux Driver
  - AXI Bus
    - AXI Interface
  - Algorithm HDL
  - Prog. Logic

Production

- ARM
  - Algorithm C
  - Linux Driver
  - AXI Bus
    - AXI Interface
  - Algorithm HDL
  - System Code
    - IP1
    - IP2
    - IP3
  - Prog. Logic

Tools:
- Embedded Coder
- HDL Coder
- Vivado
Workflow to go from an idea to a product can be broken into 5 activity types:

- Aim to establish a natural connection between all these different stages in the product development workflow
  - Enables new ideas to flow from concept into the product, without undue hindrance
  - Promotes innovation and enables agility
Q&A