Radio Reciever Mixer Model for Event Driven Simulators to support Functional Verification of RF-SOC Wireless Links

Jonathan David, SMIEEE, Sr Staff Engr, Qualcomm
IEEE BMAS 2010
Agenda

- Background and Motivation
- Mixer theory
  - Ugly Math
  - Pretty result
- Model code
- Test code
- Results
Drivers for Consumer Device IC’s

- Wires – Weight – Wallet
  - Consumers like Simplicity,
    - Convenience, Affordability
  - Cables
  - Power Outlet
  - Batteries

- Wireless connectivity is improvement.
  - If LOW power and LOW cost.
  - Lower Power -> more digital & simpler analog[Horowitz] -> smaller process nodes
  - Lower Cost -> more integration

  - RF functional verification is key enabler to success.
Using real number modeling – models for all blocks are fairly straightforward except the RX mixer.

- RF I & Q values must be rotated to BB I & Q values based on phase difference between Carrier and Local Oscillator.
Representing RF signals

\[ S_{RF} = re\{[I(t) + jQ(t)]e^{j\omega t}\} \]

\[ S_{RF} = I(t)\cos(\omega t) - Q(t)\sin(\omega t) \]

- I, Q, f [ = \omega/2\pi] are sufficient to represent most RF signals.
  - FM: I = signal Ampl, Q = 0, variation in f carries signal information
  - AM: I = signal, Q = 0, f is carrier frequency
  - QAM: I & Q carry Baseband signal, f is channel frequency.

- System Level simulations (Matlab)
  - assume fixed frequency,
  - use complex number for I & Q.
  - Assumption doesn’t hold for functional verification.
Mixer Theory

“At the core of all mixers presently in use is a multiplication of two signals in the time domain. The fundamental usefulness of multiplication may be understood from examination of the following trigonometric identity:

\[(A \cos \omega_1 t)(B \cos \omega_2 t)\]

\[= \frac{AB}{2} \left[ \cos(\omega_1 - \omega_2)t + \cos(\omega_1 + \omega_2)t \right]\]

Tom Lee [15]
The Ugly Math – Quadrature mixer

1: Get an instantaneous value for frequency (phase)

\[ \omega t = \int_{0}^{t} \omega \, dt = \omega \int_{0}^{t} dt \]

\[ \phi = \int_{0}^{t} \omega \, dt = 2\pi \int_{0}^{t} F \, dt \]

2: This makes our Mixer Equation:

\[ (A \cos \phi_{RF})(\cos \phi_{LO}) = \frac{A}{2} [\cos(\phi_{RF1} - \phi_{LO}) + \cos(\phi_{RF} + \phi_{LO})] \]

3: define a few other terms

\[ \Delta \phi = \phi_{RF} - \phi_{LO} \quad \& \quad \Sigma \phi = \phi_{RF} + \phi_{LO} \]

\[ (I \cos \phi_{RF} - Q \sin \phi_{RF})(\cos \phi_{LO}) = \frac{I}{2} M'_{11} - \frac{Q}{2} M'_{12} \]

\[ (I \cos \phi_{RF} - Q \sin \phi_{RF})(-\sin \phi_{LO}) = -\frac{I}{2} M'_{21} + \frac{Q}{2} M'_{22} \]

\[ M'_{11} = \cos(\Delta \phi) + \cos(\Sigma \phi) \]
\[ M'_{12} = \sin(\Delta \phi) + \sin(\Sigma \phi) \]
\[ M'_{21} = -\sin(\Delta \phi) + \sin(\Sigma \phi) \]
\[ M'_{22} = \cos(\Delta \phi) - \cos(\Sigma \phi) \]

4: Dropping the summing terms due to filtering

Our Quadrature receiver gives us two output signals:

\[ I_{REC} = \frac{I}{2} [\cos(\Delta \phi)] - \frac{Q}{2} [\sin(\Delta \phi)] \]
\[ Q_{REC} = \frac{I}{2} [\sin(\Delta \phi)] + \frac{Q}{2} [\cos(\Delta \phi)] \]
The mixer model only needs to know the difference in the phase between the RF frequency and LO frequency to calculate the Baseband output of the mixer.

Since integration is linear, the difference in phase can be calculated by integrating the difference in frequency.

\[
R_I = G_I [I(t) \cos(\Delta \phi) - Q(t) \sin(\Delta \phi)]
\]
\[
R_Q = G_Q [I(t) \cos(\Delta \phi - \pi/2) - Q(t) \sin(\Delta \phi - \pi/2)]
\]
always @LOin_Freq_inph, RXin_freq, RXin_linph, RFin_lquad) begin // update events
    // integrate DeltaF to get DeltaPhase
    DeltaPhi = DeltaPhi + (RXin_freq_last - LOin_Freq_inph_last)*dt*two_pi;
    NDelt = DeltaPhi/two_pi; // get 2*pi units
    DeltaPhi = DeltaPhi - NDelt*two_pi; // subtract them
end
RXin_freq_last = RXin_freq; // update the history vars
LOin_Freq_inph_last = LOin_Freq_inph;
time_last = $realtime;
Model Code – core equations

// matrix factors
M11 = $qc\_cos(\ Del\ t\ aPhi\ )\ *\ Eff\_gain\ n;
M12 = -1.0 * $qc\_sin(\ Del\ t\ aPhi\ )\ *\ Eff\_gain\ n;
M21 = signQ * $qc\_sin(\ Del\ t\ aPhi\ )\ *\ Eff\_gain\ n;
M22 = signQ * $qc\_cos(\ Del\ t\ aPhi\ )\ *\ Eff\_gain\ n;
BB_I_Iout = M11 * RXin_Iinph + M12 * RFin_Iquad;
BB_Q_Iout = M21 * RXin_Iinph + M22 * RFin_Iquad;
end
endmodule
## Testing the model – 10 cases

<table>
<thead>
<tr>
<th>TestID</th>
<th>Description</th>
<th>Frf</th>
<th>Flo</th>
<th>Fbb</th>
<th>LO IleadsQ</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&quot;Frf = Flo&quot;</td>
<td>2.4G</td>
<td>2.4G</td>
<td>2M</td>
<td>T</td>
<td>3dB</td>
</tr>
<tr>
<td>2</td>
<td>&quot;Frf &lt; Flo&quot;</td>
<td>2.4G</td>
<td>2.4005G</td>
<td>2M</td>
<td>T</td>
<td>3dB</td>
</tr>
<tr>
<td>3</td>
<td>&quot;Frf &lt;&lt; Flo&quot;</td>
<td>2.4G</td>
<td>2.404G</td>
<td>2M</td>
<td>T</td>
<td>3dB</td>
</tr>
<tr>
<td>4</td>
<td>&quot;Frf &gt; Flo&quot;</td>
<td>2.4005G</td>
<td>2.4G</td>
<td>2M</td>
<td>T</td>
<td>3dB</td>
</tr>
<tr>
<td>5</td>
<td>&quot;Frf &gt;&gt; Flo&quot;</td>
<td>2.402G</td>
<td>2.4G</td>
<td>2M</td>
<td>T</td>
<td>3dB</td>
</tr>
<tr>
<td>6</td>
<td>&quot;Frf = -Flo&quot;</td>
<td>2.4G</td>
<td>2.4G</td>
<td>2M</td>
<td>F</td>
<td>3dB</td>
</tr>
<tr>
<td>7</td>
<td>&quot;Frf &lt; -Flo&quot;</td>
<td>2.4G</td>
<td>2.4005G</td>
<td>2M</td>
<td>F</td>
<td>3dB</td>
</tr>
<tr>
<td>8</td>
<td>&quot;Frf &lt;&lt; -Flo&quot;</td>
<td>2.4G</td>
<td>2.404G</td>
<td>2M</td>
<td>F</td>
<td>3dB</td>
</tr>
<tr>
<td>9</td>
<td>&quot;Frf &gt; -Flo&quot;</td>
<td>2.4005G</td>
<td>2.4G</td>
<td>2M</td>
<td>F</td>
<td>3dB</td>
</tr>
<tr>
<td>10</td>
<td>&quot;Frf &gt;&gt; -Flo&quot;</td>
<td>2.402G</td>
<td>2.4G</td>
<td>2M</td>
<td>F</td>
<td>3dB</td>
</tr>
</tbody>
</table>
Testing the model - expected results

\[ F_{\text{expect}} = \text{abs}(F_{\text{Fr}} + F_{\text{bb}} - F_{\text{lo}}); \]

\[ I_{\text{leadsQexpect}} = \text{LO}_I \text{leadsQ} \sim (F_{\text{Fr}} + F_{\text{bb}} > F_{\text{lo}}); \]

\[ B_{\text{B}}_I_{\text{freq}} = \text{RX}_\text{BBMon}.xI_{\text{Freq}}; \]
\[ B_{\text{B}}_Q_{\text{freq}} = \text{RX}_\text{BBMon}.xQ_{\text{Freq}}; \]
\[ I_{\text{QPhasing}} = \text{RX}_\text{BBMon}.I_{\text{leadsQ}}; \]
\[ F_{\text{FromBBmonI}} = \text{RX}_\text{BBMon}.xI_{\text{PeakAc}}; \]
\[ F_{\text{FromBBmonQ}} = \text{RX}_\text{BBMon}.xQ_{\text{PeakAc}}; \]
\[ \text{PeakI nput} = \text{RXsrc}.\text{peak}; \]
\[ \text{Radius} = \text{sqrt}(\text{FromBBmonI}^2 + \text{FromBBmonQ}^2); \]

\[ \text{if}(\text{PeakI nput} > 0.0) \] begin
\[ \text{Actual Gain} = (20.0) \times \text{log10}(\text{Radius} / \text{PeakI nput}); \quad // \text{Gain in dB.} \]
\[ \text{GainOK} = (\text{abs}(\text{Actual Gain} - \text{RxGain}[\text{TestID}]) < 2.0); \]
end
Testing the model – Did we pass (Verilog)

test_passed = (I_leads_Qexpect == RX_BBMon.I_leads_Q)
          && ($qc_abs(BB_Ifreq - Fexpect) < 2e3)
          && ($qc_abs(BB_Qfreq - Fexpect) < 2e3)
          && GainOK;

$strobe("% @ %: RXTEST%02d % Gain: %g dB, Frf: %g Hz,
       Flo: %g LO: %g Hz, Fbb: %g Hz, BB: %s",
       test_passed?"SPECINFO":"SPECFAIL" , $realtime,
       TestID,
       test_passed?" Passed :":" Failed :",
       Actual Gain, $qc_abs(Frf), $qc_abs(Flo),
       LO_IleadsQ? "I leads Q": "Q leads I",
       $qc_abs(Fexpect),
       IQPhasing? "I leads Q": "Q leads I");
if (!test_passed) QcAssertFailIncrement;
Automated Test Results

SPECINFO @ 15388.000 ns: RXTEST01 Passed : Gain:3.00487 dB, Frf: 2.4e+09 Hz, Flo: 2.4e+09 LO:I leads Q Hz, Fbb: 2e+06 Hz, BB:I leads Q
SPECINFO @ 33228.000 ns: RXTEST02 Passed : Gain:3.00867 dB, Frf: 2.4e+09 Hz, Flo: 2.4005e+09 LO:I leads Q Hz, Fbb: 1.5e+06 Hz, BB:I leads Q
SPECINFO @ 47898.000 ns: RXTEST03 Passed : Gain:3.00355 dB, Frf: 2.4e+09 Hz, Flo: 2.404e+09 LO:I leads Q Hz, Fbb: 2e+06 Hz, BB:Q leads I
SPECINFO @ 60648.000 ns: RXTEST04 Passed : Gain:2.99961 dB, Frf: 2.4005e+09 Hz, Flo: 2.4e+09 LO:I leads Q Hz, Fbb: 2.5e+06 Hz, BB:I leads Q
SPECINFO @ 70528.000 ns: RXTEST05 Passed : Gain:3.00598 dB, Frf: 2.4e+09 Hz, Flo: 2.4e+09 LO:Q leads I Hz, Fbb: 2e+06 Hz, BB:Q leads I
SPECINFO @ 85398.000 ns: RXTEST06 Passed : Gain:3.00598 dB, Frf: 2.4e+09 Hz, Flo: 2.4e+09 LO:Q leads I Hz, Fbb: 2e+06 Hz, BB:Q leads I
SPECINFO @ 103228.00 ns: RXTEST07 Passed : Gain:3.00735 dB, Frf: 2.4e+09 Hz, Flo: 2.4005e+09 LO:Q leads I Hz, Fbb: 1.5e+06 Hz, BB:Q leads I
SPECINFO @ 117908.00 ns: RXTEST08 Passed : Gain:3.00546 dB, Frf: 2.4e+09 Hz, Flo: 2.404e+09 LO:Q leads I Hz, Fbb: 2e+06 Hz, BB:I leads Q
SPECINFO @ 130668.00 ns: RXTEST09 Passed : Gain:3.0074 dB, Frf: 2.4005e+09 Hz, Flo: 2.4e+09 LO:Q leads I Hz, Fbb: 2.5e+06 Hz, BB:Q leads I
SPECINFO @ 140548.00 ns: RXTEST10 Passed : Gain:2.98868 dB, Frf: 2.402e+09 Hz, Flo: 2.4e+09 LO:Q leads I Hz, Fbb: 4e+06 Hz, BB:Q leads I

*  
*  
*  **********  
*  **********  
*  *******  
**  
Simulation PASSED
Example Waveforms Flo > Frf
Lissajou plot $\text{Flo} < \text{FrF}$
Summary

- Passband model is conceptually simpler:
  - Single real value sampled at 8-20x RF carrier
  - Mixer is simple multiplication with approx sinusoid Quad LO signal.
  - Followed by 3rd or 4th order discrete time IIR filter.
- This model is approx 100x faster in simulation than Passband model
- Inserting RF signal path in chip-chip simulations is only 2-3x longer run times vs all digital signal path (with DAC-->ADC channel model)
  - This Run time is acceptable given ability to check FULL functionality of Radio control and data interfaces.