

四个步骤快速投入生产： 针对软件定义无线电使用基于模型的设计

第一部分 — ADI/Xilinx SDR 快速原型制作平台： 功能、优势以及工具

作者：Di Pu、Andrei Cozma和Tom Hill

摘要

无线系统的概念与设计实现之间存在巨大的差异。要缩小这种差异通常都要涉及到几组来自各领域的工程师团队（比如RF、SW、DSP、HDL和嵌入式Linux[®]），并且很多情况下项目在开发的早期阶段便由于难以协调各设计团队而偏离了原先计划。

本系列文章分为四个部分，将讨论平台和工具的进步；这些技术进步允许开发人员快速进行无线系统的仿真与原型制作，同时建立与保持投入生产的可实现路径。作为实际过程的一个示例，我们将对无线SDR平台进行原型制作，该平台可接收并解码自动相关监视广播（ADS-B）信号，以便我们检测并汇报附近飞行中的商用飞机的位置、高度和速度。本例中需用到MATLAB[®]和Simulink，以及集成和内嵌硬件/软件的技能。硬件平台将采用ADI/Xilinx[®]软件定义无线电（SDR）原型制作系统。使用MATLAB和Simulink[®]后，将执行下列任务：

- 设计用来解码ADS-B消息的信号处理算法
- 仿真接收ADS-B信号的RF接收器
- 生成C语言代码和HDL代码
- 以目标收发器和FPGA上的记录数据和实时数据验证HDL代码

最终得到一个能够在适于生产的硬件上实现的可行RF SDR设计，我们将把它带到当地机场来验证性能和功能。

该文章系列有四部分，第一部分将讨论ADI/Xilinx SDR原型制作系统、该系统的功能和优势，并简要描述工具流程。第二部分将回顾自动相关监视广播（ADS-B）信号，并解释仿真时如何在MATLAB和Simulink中解码它们的信息。第三部分将讨论和演示如何使用环路中硬件（HIL）并以目标收发器捕捉信号，并同时在Simulink的主机中进行信号处理以便验证。第四部分将说明如何通过第二部分开发并在第三部分验证的算法，并使用MathWorks的HDL编码器以及嵌入式编码器生成代码，然后在生产硬件中进行部署；最后，我们将在机场采用真正的ADS-B信号来操作平台。

简介

随着人们对通信方式和手段的需求呈指数式增长，方便而经济高效地修改无线电设备也随之成为一项重要的业务。在这样的背景下，软件定义无线电技术最近得到了广泛的部署，因为这项技术推动通信向着灵活、高性价比、功能更强大的方向发展¹。SDR系统的目的是在软件和可重复编程逻辑中部署尽可能多的调制/解调和数据处理算法，以便通信系统能够仅通过更新软件和可重复编程逻辑而轻松进行再配置，并且无需更改硬件平台。

随着片上系统（SoC）的出现（比如集CPU的灵活性与FPGA的处理能力于一体的Xilinx Zynq[®] All Programmable SoC），设计人员终于能够将SDR系统的数据处理功能和其它处理任务纳入单个器件中。数据调制/解调算法等处理密集型任务分流至器件的可编程逻辑，而数据解码和渲染、系统监控和诊断以及用户界面等任务延缓至处理单元进行。

同时，无线系统的原型制作数十年来一直是个争论不休的话题，而最近几年才刚出现针对FPGA的完整设计流程 — 从模型创建到完整实施 — 这要归功于像MathWorks的MATLAB和Simulink等建模和仿真工具的革命。无线系统的原型制作正在改变工程师和科学家工作的方式，它将设计任务从实验室和现场带到了桌面上²。现在，工程师可以对整个无线系统（比如SDR系统）进行建模，从而可观察系统的表现，并在现场实际实施之前进行调节。这样做有很多好处，比如加快系统集成、减少对设备的依赖。此外，完成SDR系统的Simulink模型之后，C语言代码和HDL代码可自动生成，然后部署到Zynq SoC上，从而节省时间并避免手动编码错误。将系统模型链接到快速原型制作环境可进一步降低风险，因为后者允许SDR系统在实际条件下运作。

该文章系列有四部分，第一部分将讨论ADI/Xilinx SDR快速原型制作系统、该系统的功能和优势，并简要描述工具流程。文章解释了ADI RF IC技术和参考设计硬件与软件如何降低设计技巧，从而使客户减轻风险，并缩短产品上市时间。

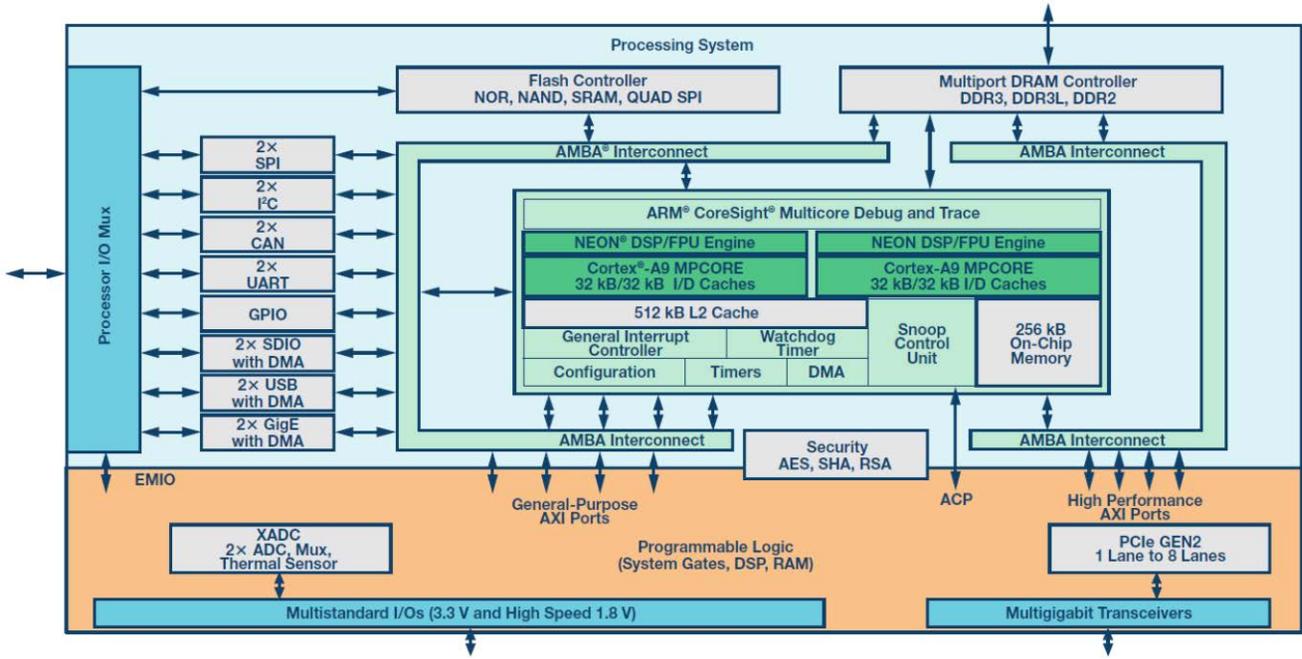


图1. Xilinx Zynq SoC功能框图

用于SDR的Zynq

当需要执行数据处理、通信和用户界面等具有不同处理带宽要求和实时限制的任务组合时，需要用到高级SDR系统。为了实现这样的系统，所选的硬件平台必须鲁棒且可扩展，同时还需为将来的系统改进和扩张创造条件。Xilinx Zynq-7000 All Programmable SoC满足这些要求，提供高性能处理系统以及可编程逻辑，如图1所示³。可编程逻辑配上处理系统便具有了出色的并行处理能力、实时性能、快速计算能力以及连接的多样性。

Zynq SoC的处理系统包括一个双核ARM® Cortex®-A9处理器和一个NEON协处理器，以及多个用于加速软件执行的浮点扩展单元。为了完全发挥系统的能力，可以在双核ARM处理器上采用嵌入式Linux或实时操作系统。该处理器是自足的，可在不配置可编程逻辑的情况下使用，这点对于软件开发人员来说很重要，他们可以与设计FPGA结构的硬件开发人员同时进行代码的开发工作。

在可编程逻辑方面，该器件拥有多达444,000个逻辑单元和2,200个DSP Slice，可提供巨大的处理带宽，允许Zynq设备应对各种充满挑战的信号处理应用。5个高吞吐速率的

AMBA®-4 AXI高速互连可将编程逻辑紧密地耦合到处理系统，提供相当于3,000以上引脚的有效带宽⁴。

用于SDR的AD9361捷变宽带RF收发器

近年来，ADI公司将革命性的SDR产品推向了市场，以支持不断变化的SDR要求和系统架构。在这方面，ADI最重要的产品包括AD9361/AD9364集成式RF捷变收发器。AD9361 (2×2)⁵和AD9364 (1×1)⁶是用于SDR架构的高性能、高度集成的RF收发器IC，适合无线通信基础设施、防务电子系统、RF测试设备和仪器，以及通用软件定义无线电平台等应用。这些器件集RF前端与灵活的混合信号基带部分为一体，集成频率合成器，为处理器或FPGA提供可配置数字接口，从而简化设计导入。这些芯片工作频率范围为70 MHz至6 GHz，涵盖大部分特许执照和免执照频段，通过对AD9361和AD9364器件编程可改变采样速率、数字滤波器和抽取参数，使该芯片支持的通道带宽范围为低于200 kHz至56 MHz⁷。图2显示了AD9361器件的功能框图。

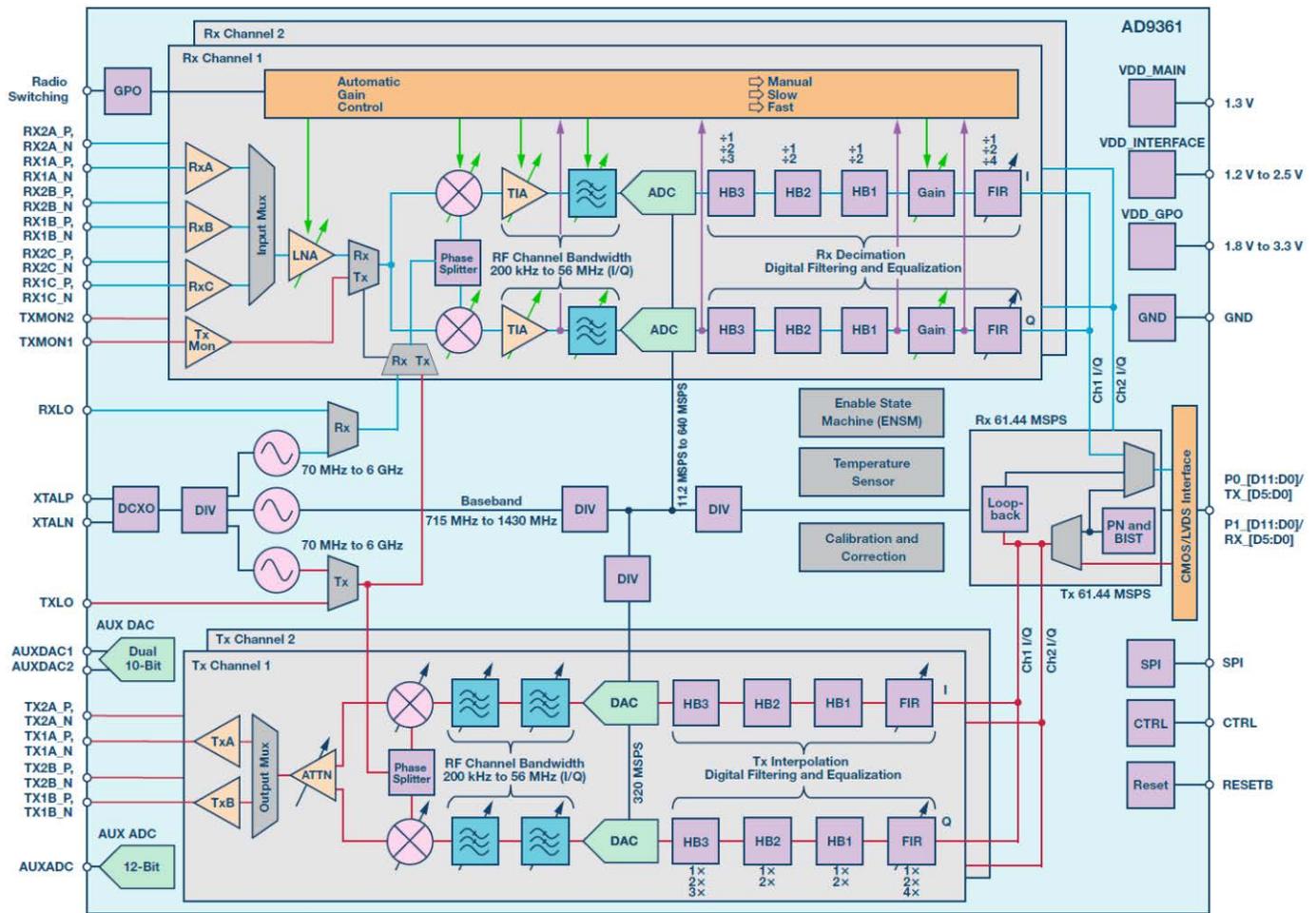


图2. AD9361 功能框图

为了帮助客户缩短产品上市时间并减轻整体开发负担，ADI公司更进一步，提供的SDR解决方案可用于具有无缝FPGA连接能力的完整生态系统中，为完整的无线电系统设计提供快速原型制作和开发环境。AD-FMCOMMSx-EBZ快速开发和原型制作板属于高速模拟FMC模块，集成AD9361或AD9364捷变RF收发器IC或分立式信号链，可无缝连接Xilinx FPGA开发平台生态系统。这些板可完全通过软件定制，无需更改任何硬件，提供可供下载的Linux驱动程序和裸机软件驱动程序、原理图、电路板布局文件和有助于设计的参考材料，可前往ADI的Wiki知识库获取。表1总结了不同FMCOMMSx平台的产品特性。

表1. FMCOMMSx平台

平台	产品特性
AD-FMCOMMS5-EBZ	该SDR快速原型制作板集成两个AD9361 2 × 2捷变收发器IC，能够为4个接收器通道和4个发送器通道提供完全同步能力，可创建4 × 4 MIMO系统的任意子系统。支持宽带70 MHz至6 GHz和2.4 GHz调谐端口。AD-FMCOMMS5-EBZ资源页面（wiki知识库）： http://wiki.analog.com/resources/eval/user-guides/ad-fmcomms5-ebz
AD-FMCOMMS4-EBZ	该1 × 1 SDR快速原型制作板集成AD9364捷变RF收发器IC，可经软件配置在2400 MHz至2500 MHz范围内实现最高RF性能，或者可经软件配置工作在AD9364的70 MHz至6 GHz完整RF调谐范围内，以进行系统原型制作和开发。AD-FMCOMMS4-EBZ资源页面（wiki知识库）： http://wiki.analog.com/resources/eval/user-guides/ad-fmcomms4-ebz
AD-FMCOMMS3-EBZ	该2 × 2型号的SDR快速原型制作板集成AD9361捷变RF收发器IC，支持AD9361的70 MHz至6 GHz全RF调谐范围。该套件非常适合寻求具有宽调谐能力的统一开发平台的无线通信系统架构师。AD-FMCOMMS3-EBZ资源页面（wiki知识库）： http://wiki.analog.com/resources/eval/user-guides/ad-fmcomms3-ebz
AD-FMCOMMS2-EBZ	该2 × 2 SDR快速原型制作板集成AD9361捷变RF收发器IC，经调谐可在2400 MHz至2500 MHz范围内实现最高RF性能。该套件适合寻求最优系统性能，并要求在RF频谱定义范围内满足AD9361数据手册中规格的工程师使用。AD-FMCOMMS2-EBZ资源页面（wiki知识库）： http://wiki.analog.com/resources/eval/user-guides/ad-fmcomms2-ebz

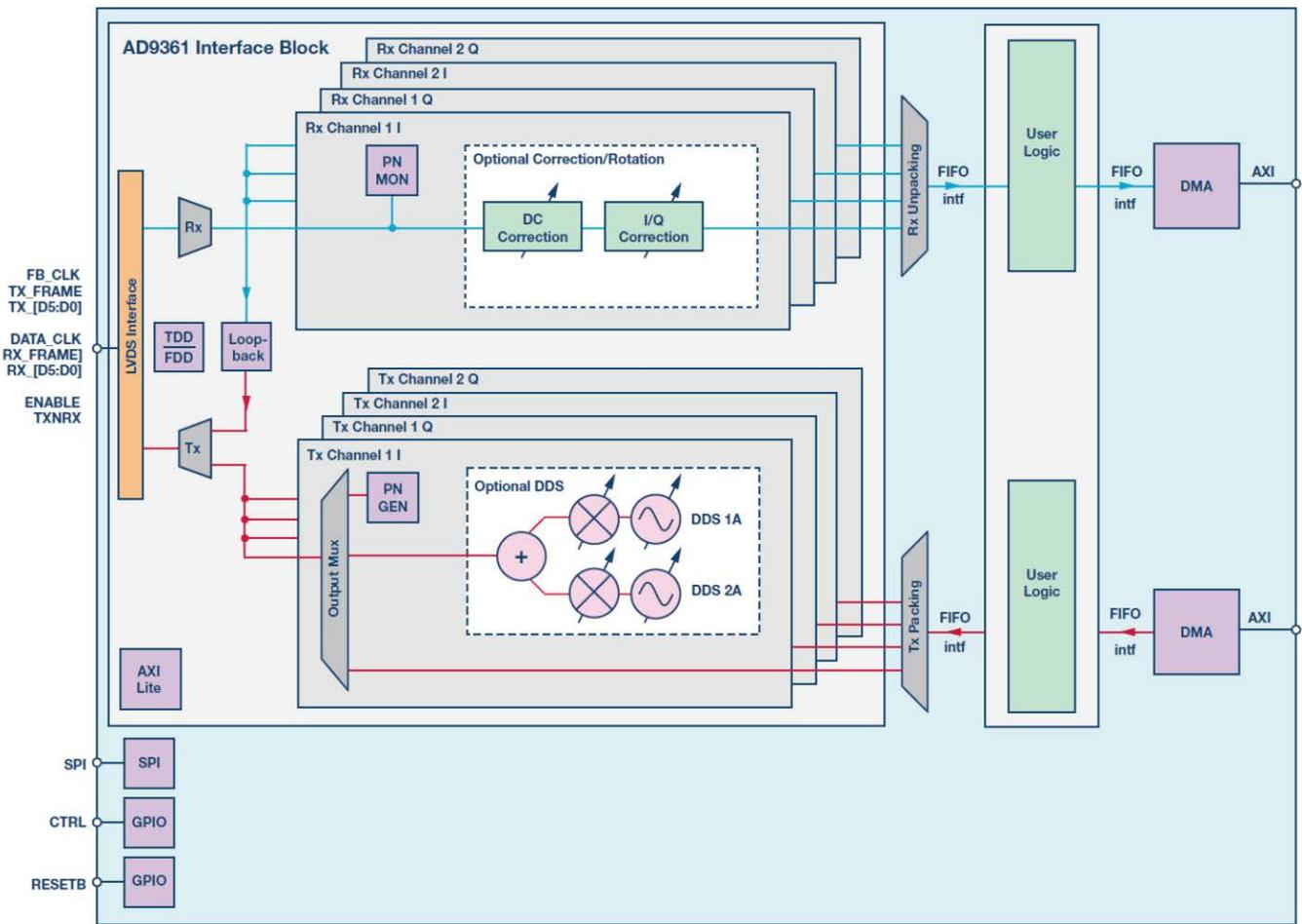


图3. ADI HDL 和软件基础架构

Zynq SDR快速原型制作平台

参考设计

ADI公司与FMCOMMSx平台一同提供完整的Vivado框架, 采用Linux和裸机软件基础架构, 可同时用于原型制作以及最终生产系统中。图3显示了ADI支持FMCOMMSx板的Zynq基础架构。

该高级框图说明了ADI参考设计在Xilinx Zynq SoC上是如何划分的。一路HDMI输出用于在监视器上显示Linux界面, 鼠标和键盘可通过USB 2.0端口连接到系统。ARM Cortex A9处理系统运行ADI公司提供的Ubuntu Linux, 其中包括: 与ADI公司FMCOMMS硬件接口所需的Linux IIO驱动程序, 用于监测和控制的IIO Oscilloscope (示波器)⁸用户空间应用程序, 支持实时数据采集和通过TCP控制系统的libiio服务器⁹, 在远程计算机上运行的客户端, 以及整合嵌入式编码器所生成C语言代码的可选用户应用程序, 可用于控制器的Simulink模型。

软件基础架构

所有ADI Linux驱动程序均基于Linux工业I/O (IIO) 子系统, 现已包含在所有主流Linux内核中。IIO Scope是ADI公司开发的一款开源Linux应用程序, 运行在Xilinx Zynq中的双核ARM Cortex-A⁹的内核上, 能够显示连接到Xilinx Zynq平台的ADI FMC卡所获取的实时数据。这些数据可以在时域中、频域中或以星座图的形式显示。支持以不同的常用文件格式

(如逗号分隔值或.mat MATLAB数据文件等) 保存所捕获的数据以供进一步分析。IIO Scope提供一个图形用户界面, 用于更改或读取ADI FMC卡的配置。libiio服务器支持实时数据采集、通过传输控制协议 (TCP) 控制系统以及运行于远程计算机上的客户端¹⁰。服务器运行于Linux下的嵌入式目标上, 通过TCP管理目标与远程客户端之间的实时数据交换。此库抽取了硬件的低级详情, 提供简单但完整的编程接口, 可用于高级项目。它的模块化架构、设计良好的API以及内置的网络功能允许用户创建应用, 而这些应用不仅能运行在IIO设备已连接的系统中, 还能运行在通过网络远程连接的系统中。它首先针对Linux, 不过现在也能通过库的远程后端支持Windows。它以C语言写成, 并授权给LGPL, 结合了C#、Python和MATLAB的特点。MathWorks IIO客户端¹¹可以作为系统对象集成到MATLAB和Simulink原生应用程序中。它设计用来配合连接FPGA/SoC平台 (运行ADI Linux发布版) 的以太网进行数据交换, 可让MATLAB或Simulink模型执行下列功能:

- 发送数据流至目标, 并从目标接收数据流
- 控制目标的设置
- 监控不同的目标参数

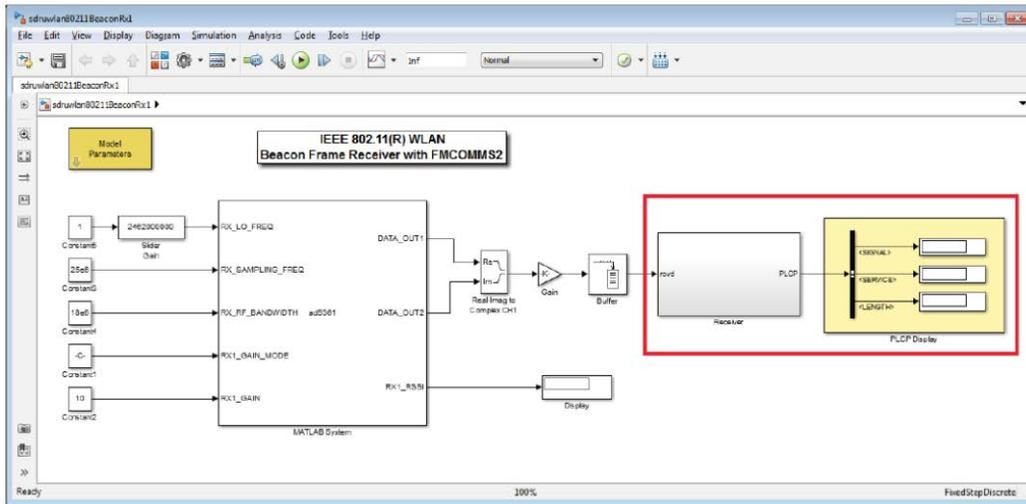


图4. 信标帧接收器示例的屏幕截图

MATLAB和Simulink均提供IIO系统对象，具体取决于用户从MATLAB脚本中对其进行调用，还是将其整合至MATLAB系统模块中。ADI提供针对FMCOMMS平台的Linux软件和HDL基础架构，配合MathWorks和Xilinx提供的工具后，便是进行SDR应用原型制作的绝佳环境，并且它还包含可随时投入生产的组件，这些组件可以集成至SDR系统——有助于缩短从概念到生产所需的时间并降低成本。

为了帮助客户快速而轻松地掌握IIO系统对象，我们提供了几个基于该界面的MATLAB和Simulink示例，比如信标帧接收器¹²、QPSK发射器和接收器¹³，以及LTE发送器和接收器¹⁴。在这些示例中，FMCOMMSx平台由IIO系统对象配置，并用作RF前端并无线发送或接收模拟信号。这些信号通过IIO系统对象以数据流的形式发送至目标，或从目标接收。所有其它信号处理均在MATLAB或Simulink中执行。图4是信标帧接收器示例的屏幕截图，该图显示了IIO系统对象和其它Simulink模块之间的典型连接。

MathWorks对Zynq的支持

MathWorks支持基于Zynq的SDR，具体表现在以下四个方面：

1. AD9361 Simulink模型

由于AD9361是一款集成式RF收发器芯片，信号探测和内部工作监控是不太现实的。因此，MathWorks和ADI合作开发了AD9361的SimRF™模型，可对芯片的工作进行仿真，以便客户能够真正了解到这其中发生了什么，并知晓在现实中难以重现的不同测试条件下芯片性能如何。SimRF使用相同的基带或电路包络模块——比如放大器、混频器和S-参数模块——提供RF系统设计中的组件库和仿真引擎。它是适合AD9361 RF收发器建模的有效工具。系统级AD9361捷变RF收发器模型（如图5所示）精确重现了AD9361的功能，并以MathWorks硬件支持包的形式向用户提供¹⁵。

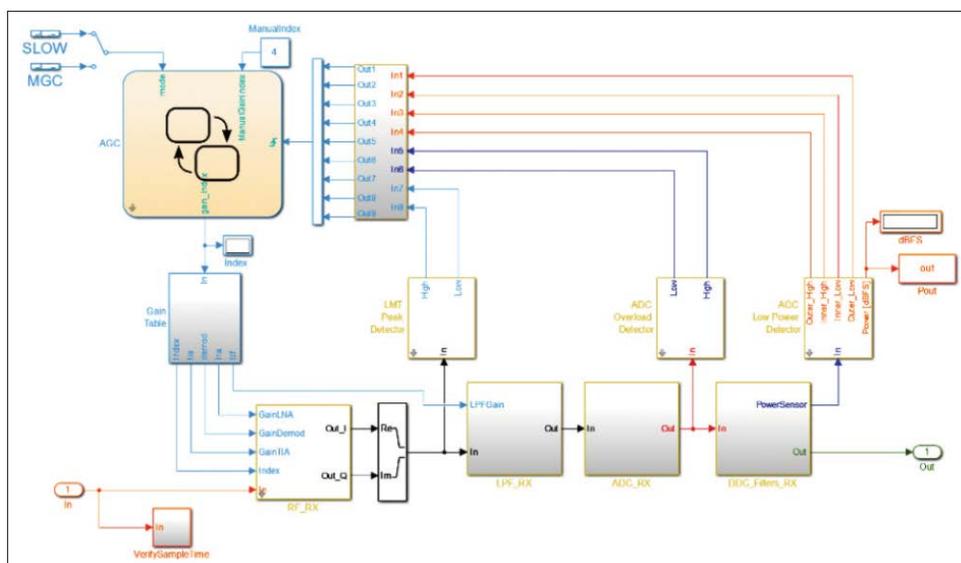


图5. AD9361 捷变RF接收器的MathWorks SimRF模型

SimRF模型已在实验室中经过了功率频谱测量验证。不同频率和功率水平下的收发器噪声和非线性特性也已识别。然后，这些模型设计为生成相同的特性，并在设计范围内得到验证。

采用AD9361收发器SimRF模型后，用户可以：

- 预测RF缺陷对测试信号的影响
- 使用参考音和LTE信号
- 生成或导入测试向量，并评估非线性、噪声、增益和相位不平衡、频谱泄露以及其它RF发送器和接收器缺陷导致的影响
- 加入干扰信号并评估时域或频域结果

2. 通信和DSP系统工具箱功能

MathWorks产品 — 比如*Communications System Tool-box*^{TM16}、*Signal Processing Toolbox*^{TM17}、*DSP System Toolbox*^{TM18}和SimRF¹⁹ — 具有业界标准算法和应用程序，可进行SDR系统的系统性分析、设计与调谐。所有这些工具均提供了创建高保真SDR模型的途径，可在进行真实物理部署前用来验证通信系统的表现和性能。

3. 用于Zynq的Simulink工作流程

MathWorks的MATLAB和Simulink是用于多领域仿真和基于模型设计的环境，非常适合仿真具有通信算法的SDR系统。通信算法调节增益、频率偏移、时序偏移和其它性能变量，并经常能更好地同步发送器和接收器系统。利用仿真评估通信算法可以有效地确定SDR设计是否合适，判断其合适后再进行昂贵的硬件测试，从而减少算法开发的时间和成本。图6给出了设计通信算法的有效工作流程，步骤如下：

- 使用基于模型的设计环境提供的库构建精确的SDR模型
- 仿真系统行为以验证系统表现是否符合预期
- 产生C代码和HDL进行实时测试和实施
- 利用原型制作硬件测试通信算法

在原型制作硬件上进行仿真和测试后，如果SDR系统的性能证明是令人满意的，那么在最终生产系统上实施与部署系统也将是安全的。

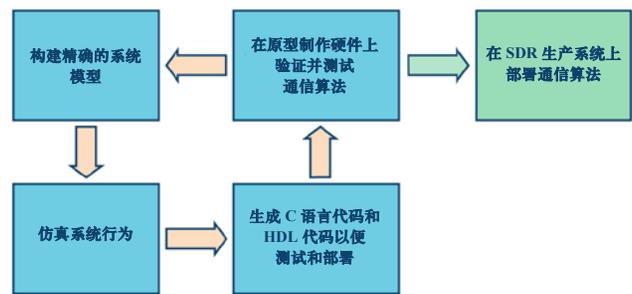


图6. 通信算法设计的工作流程

4. Simulink平台集成至Zynq SDR套件

一旦SDR系统使用MathWorks的Embedded Coder^{®20}和HDL Coder^{TM21}等工具完成了全面的验证，用户就可以利用嵌入式编码器和VHDL或使用HDL编码器的Verilog来生成C语言代码，然后将代码部署到原型制作硬件上进行测试，之后便可进入最终生产系统。此时应指定软件和硬件实施要求，如定点和时序行为。自动生成代码有助于缩短从概念到实际系统实施所需的时间，消除手动编程错误，确保实际SDR实施与模型相符。图7给出了在Simulink中进行SDR系统建模并将其转移到基于Xilinx Zynq SoC的最终生产系统所需的实际步骤。

第一步是在Simulink中对SDR系统进行建模和仿真。在这一阶段，通信算法被划分为在软件中实现的模块和在可编程逻辑中实现的模块。划分和仿真完成后，利用嵌入式编码器和HDL编码器将SDR模型转换为C语言代码和HDL代码。基于Zynq的原型制作系统用来验证通信算法的性能，并且帮助进一步调谐SDR模型，然后转移到实际生产阶段。在生产阶段，将自动生成的C代码和HDL集成到复杂的生产系统框架中。此工作流程确保通信算法在到达生产阶段之前经过全面验证和测试，使得系统鲁棒性具有高可信度。Zynq针对嵌入式编码器和HDL编码器推出的硬件支持包提供集成式硬件/软件设计、仿真和验证框架，将基于模型的设计集成至工作流程中，简化了Zynq平台的编程，并实现了快速设计迭代周期，同时有助于尽早检测和纠正设计以及规格错误²²。

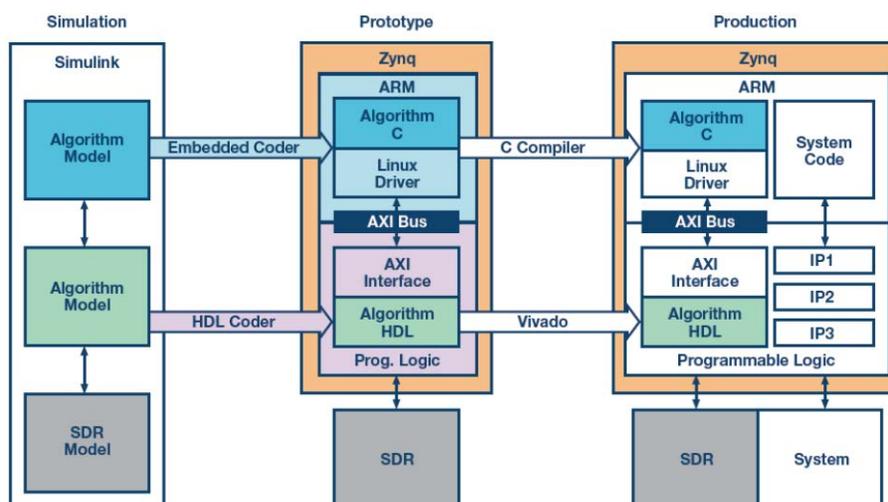


图7. 从仿真到生产的过程

结论

本文说明了现代SDR系统的要求和趋势，以及为满足这些要求和帮助实现更高性能SDR解决方案而由MathWorks、Xilinx和ADI公司带给市场的工具和系统。通过将MathWorks基于模型的设计和自动生成代码工具与强大的Xilinx Zynq SoC和ADI集成式RF收发器结合，SDR系统设计、验证、测试和实现可以比以前更有效率，进而提高无线电系统性能并缩短产品上市时间。ADI的FMCOMMS平台搭配Avnet Zynq-7000 AP SoC便可提供强大的原型制作环境，供采用MathWorks MATLAB和Simulink的SDR算法设计使用。FMCOMMS平台带有一组开源参考设计，旨在为所有希望评估该系统的人士提供一个起点，并且帮助启动任何新SDR项目。

在本系列的下一篇文章中，我们将进一步深入探讨SDR设计过程，回顾自动相关监视广播 (ADS-B) 信号特性，并解释如何在MATLAB/Simulink仿真中解码这些信息。

有关本文中所涉及话题、文档、视频和参考设计的更多信息，请查看“参考文献”部分。

参考文献

- ¹ “什么是软件定义无线电？”无线创新论坛。
- ² 基于模型的设计。MathWorks。
- ³ Zynq-7000 All Programmable SoC。Xilinx。
- ⁴ Hill, Tom。“借助MATLAB将电机驱动迁移到Zynq SoC设计中。”Xcell杂志，第87期，2014年第二季度。

- ⁵ AD9361。
- ⁶ AD9364。
- ⁷ “ADI的软件定义无线电解决方案”。ADI公司。
- ⁸ IIO示波器。ADI公司Wiki知识库。
- ⁹ Simulink Libbio。ADI公司Wiki知识库。
- ¹⁰ 什么是Libbio? ADI公司Wiki知识库。
- ¹¹ IIO系统对象。ADI公司Wiki知识库。
- ¹² 信标帧接收器示例。ADI公司Wiki知识库。
- ¹³ QPSK发送器和接收器示例。ADI公司Wiki知识库。
- ¹⁴ LTE发送器和接收器示例。ADI公司。
- ¹⁵ AD9361。
- ¹⁶ “Communications System Toolbox”。MathWorks公司。
- ¹⁷ “Signal Processing Toolbox”。MathWorks公司。
- ¹⁸ “DSP System Toolbox”。MathWorks公司。
- ¹⁹ SimRF。MathWorks公司。
- ²⁰ “HDL Coder”。MathWorks公司。
- ²¹ “Embedded Coder”。MathWorks公司。
- ²² “Simulink支持Xilinx Zynq”。MathWorks公司。

MATLAB和Simulink是MathWorks公司的注册商标。其它商标列表请参见：www.mathworks.com/trademarks。其它产品或品牌名称可能是各自所有人的商标或注册商标。



Di Pu [di.pu@analog.com]是ADI公司的系统建模应用工程师，支持软件定义无线电平台和系统的设计与开发。她与MathWorks密切合作，共同解决最终客户的难题。加入ADI公司以前，她曾于2007年获得中国南京理工大学 (NJUST) 电气工程本科学位，并在2009和2013年获得美国马萨诸塞州伍斯特理工学院 (WPI) 电气工程硕士和博士学位。她是2013年WPI 博士论文西格玛西研究奖获得者。



Di Pu

Andrei Cozma [andrei.cozma@analog.com]是ADI公司工程设计经理，负责支持系统级参考设计的设计与开发。他拥有工业自动化与信息技术学士学位及电子与电信博士学位。他参与了电机控制、工业自动化、软件定义无线电和电信等不同行业领域的项目设计与开发。



Andrei Cozma

该作者的其它文章：

[基于FPGA的系统提高电机控制性能](#)
第49卷第1期

Tom Hill, Xilinx公司系统发生器产品经理[tom.hill@xilinx.com]Tom Hill拥有EDA行业超过18年的从业经历。Hill监管Xilinx DSP目标设计平台有关的所有产品、战略和企业营销活动。Hill最近在AccelChip公司担任技术营销经理，负责DSP应用的产品指导和高级设计方法与工具应用。加入AccelChip以前，Hill曾经担任多款FPGA和ASIC频率合成工具的产品经理、技术营销经理、技术营销工程师和现场应用工程师。Hill的职业生涯始于Allen-Bradley和Lockheed公司，担任硬件和ASIC设计工程师。Hill拥有克里夫兰州立大学电气工程学士学位。



Tom Hill